BreachEscape

Project report

Guided by: Prof.Yasin Ceran

By:
Anurag Gate
Bhakti Mehta
Harshada Kulkarni
Prachi Tamhankar
Sanjana Bothale

Table of Contents

1. Acknowledgement

We would like to express our deepest appreciation to all those who provided us the support to complete this project. We would like to convey our sincere thanks to our advisor and mentor, Professor Yasin Ceran, for his expert guidance and constant supervision throughout this independent study. We are highly indebted to him for providing his invaluable inputs for the development of the prototype.

Furthermore, we would like to thank Professor Vasu Kadambi, our capstone advisor for motivating us to choose this independent study option. His valuable insights on Machine Learning techniques made us think of new dimensions in implementing our project. We would like to convey our sincere thanks to all our Professors for investing their effort and time in guiding and teaching us throughout the MSIS program.

Finally, we are grateful to our institution, Santa Clara University, Leavey School of Business for nourishing us with knowledge and warmly welcoming us every day.

2. Executive summary

Though people around the world have reached the far point of desensitization to news stating a data breach; identifying the possible threats, analyzing the risk and then protecting the user data have become very important with stricter implementation of regulations. According to analysts, big companies like Reddit, Macy's and Bloomingdale have experienced number of breaches across the organization to join the list of breach victims. Public attention is needed in the matter of compromised data because data breaches can result in the loss of millions or even billions of private and sensitive public data affecting the organization as well as people whose personal information may have been stolen. We aim at developing an application that would help companies identify and manage potential problems that could undermine their business initiatives or projects. To carry out a Risk Analysis, identifying the possible threats and then estimating the likelihood that these threats will materialize is of prime importance.

In this project, we have built a unique website - BreachEscape, which provides analytical insights in the Data Security sector based on the parameters like location, intensity and domain of breaches. We have provided a risk calculator to understand how prone a company's data to potential breaches is, by considering factors like level severity, amount of records and type of data. This was a perfect opportunity for us to infuse our knowledge in implementing all concepts we have learnt so far and the new skills that we acquired during this project. MSIS Course at SCU being a perfect blend of technical and managerial courses has prepared us to materialize our ideas.

3. Project Background

A massive amount of data is generated, used, supplied and exchanged every single day by thousands of business proceedings all around the world. This data might be extremely crucial (financial/ personal/ national) and highly confidential. Therefore, it has become imperative to secure all this crucial data from potential hackers and cyber criminals in every way possible. No breach is completely preventable, but it can definitely be avoided if warned about it in advance. If organizations want to stay ahead of attackers and effectively prevent data breaches, a new mindset needs to be adapted. Companies are no longer just required to announce that their systems have been breached but also pay huge fines if their data is compromised. Therefore, to avoid such unwanted consequences, we have built a Data-risk analytics website - BreachEscape, to help various companies understand their data security needs by providing them analytical insights based on the parameters like location, intensity domain of breaches. We have also implemented the most sought-after feature in the industry today, a risk assessment calculator to understand how prone a company's data to potential breaches is.

We believe, this design capstone project has helped us learn new ideas & concepts and implement them in our project to get the solution for our problem. Also, the concepts learnt throughout our MSIS curriculum will help us to achieve the aim of the project. Since our team is a mix of individuals having different educational and work background, this project has been an interesting challenge where we could share the ideas from our experience and learn new things to grow as a professional.

4. Problem Statement

In today's era, data breaches have become a major problem due to the amounts of sensitive information exchanged on the internet for business as well as personal purposes on a day to day basis. Keeping track of all the confidential information and taking preventive measures to avoid any data hacking or misuse has become extremely imperative. Thus, businesses are turning to smart tech tools like breach meters to protect their data from internal or external violation by predicting weaker links in the data corpus.

Thus, we have proposed a novel website, which uses Risk Assessment Calculator and a comprehensive visualization tool to provide breach index score and detailed data insights for an organization. We believe that BreachEscape can become a major reference point for people from different strata of Information Technology and Data security. This website is easy to access and use, providing organization specific and relevant data breach information.

5. Project Objectives

We aim at providing the following features in our project Breach-Escape:

5.1 Risk Assessment Calculator:

In the last few years, several countries are enacting legislation for Breach reporting, and it is mandated in 46 states in the United States. Thus, there is more transparency and we came across a constant stream of breaches in the news. There is plenty of precedent for creating scales to help understand the severity of data breaches. Thus, identifying the need for a measurement to assess the gravity of different breaches, we came up with the idea of Risk assessment calculator. This Risk assessment calculator will help us calculate the index score of the breach and map it to a breach severity category based on the weighted values for number of records, type of data, source of the breach etc.

5.2 Analysis and Visualization:

This is one of the premium features of Breach-Escape where the user is exposed to all the analytical insights about the Data Breach scenario.

5.3 Website design:

Develop a simple and user-friendly UI for our customers, so that they can access the above-mentioned features and help them eliminate data breach risks.

6. Project Stakeholders

The BreachEscape application represents the interests of several groups listed below:

End Users:	1. Security solution providers 2. Risk Analysts 3. Organizations with sensitive data 4. BreachEscape Admin
SCU MSIS Capstone Management:	1. Professor Vasu Kadambi, Dean's Executive Professor of MSIS/Capstone Director. 2. Professor Yasin Ceran, Assistant Professor/Capstone Advisor.
SCU MSIS Capstone Team:	1. Anurag Gate 2. Bhakti Mehta 3. Harshada Kulkarni 4. Prachi Tamhankar 5. Sanjana Bothale

7. Project Staffing

Name	**Role**
Professor Yasin Ceran	SCU Capstone Advisor
Anurag Gate	SCU Capstone Team Member
Bhakti Mehta	SCU Capstone Team Member
Harshada Kulkarni	SCU Capstone Team Member
Prachi Tamhankar	SCU Capstone Team Member
Sanjana Bothale	SCU Capstone Team Member

8. Project Planning & Management Methodology

In this project we have developed a website which is linked to our python script for calculating the risk assessment score. We have used Agile methodology for our software process development.

The agile methodology decreases complexity by breaking it down into small iterations. We followed the technique as it is suitable for smaller teams with size 5-8. This has helped promote adaptive planning, early delivery, continuous improvement and rapid and flexible to change.
We have followed Agile SCRUM methodology in our project and the key characteristics are as follows:

8.1 BI-weekly meetings

We ensured that we would meet once a week, in-person and have one meeting via google hangout or WhatsApp group to work on the solutions to problems faced by any of the team members.

8.2 Weekly meetings with our Guide

We had weekly meetings with our capstone guide, Prof Yasin Ceran to receive feedback regarding the work done during the week and would take his guidance regarding the deliverable to be done next week. This was an iterative process of making changes based on the feedback received from the professor which was a key factor in using the Agile methodology.

8.3 Division of tasks

We divided the tasks based on the machine learning techniques, website design, integration using flask and tableau dashboards. We divided it into separate modules and prioritized it assigning timelines of the tasks.

8.4 Sprints (Time bound tasks)

We held meeting every week to track the progress and made sure that the deliverables were done on time.

8.5 Running the scripts and demos

We tried testing each module independently before integrating it together so that we could identify and modify the flaws at the modular level.

9. Project Timeline

Phases	Deliverables	Activities	Start Date	End Date
Project Initiation				
	Project overview	Introductory Meeting	01/22/2019	01/22/2019
		Project proposal Version 1 draft	01/22/2019	01/29/2019
		Project proposal Version 1.1	01/29/2019	02/07/2019
		Project proposal Version 1.2	02/07/2019	02/23/2019
	Project scope	Defining Project Scope	02/28/2019	03/02/2019
Project Planning				
	Requirements gathering	Functional and Nonfunctional requirements	01/29/2019	03/02/2019
	Technology Stack	Database Languages for UI design Scripting language Machine learning model	02/02/2019	03/09/2019
	Ownership/Distribution	Work distribution among team members	03/09/2019	03/09/2019

	Learning Plan	Skills and learning curve	03/02/2019	03/09/2019
Project execution				
	Design document		03/01/2019	03/09/2019
	Use Cases		03/01/2019	03/09/2019
	DFDs		03/01/2019	03/09/2019
Development/execution	**Data gathering, Cleaning and Transformations**		03/07/2019	03/21/2019
	Machine Learning		03/21/2019	04/18/2019
	UI design		04/11/2019	05/02/2019
	Dashboards		04/25/2019	05/02/2019
Project monitoring and control				
	Documentation		04/18/2019	05/23/2019
	Unit testing		04/25/2019	05/09/2019
	Functional testing		05/09/2019	05/16/2019
	End to End testing		05/16/2019	05/18/2019
Project closure				
	Project Report		05/23/2019	05/23/2019
	Incorporate feedback		06/01/2019	06/06/2019

	Final report submission		06/11/2019	06/11/2019
	Final presentation		06/11/2019	06/11/2019

10. Project Requirements

10.1 Functional requirements:

10.1.1 Process Oriented:

1. Business process 1: Customer lands on the homepage of BreachEscape Website

- The system should allow the user to land on the homepage of BreachEscape once the Customer enters URL and hits enter.
- Navigation menu with Home, Contact us, Risk calculator and Interactive Dashboards should be visible to the customer.
- System should display the Home page and the details about BreachEscape should be visible.
- System should allow the user to click on all menu items and should be navigated to the respective pages.

2. Business Process 2: Customer Registration/Log in to the BreachEscape Website

- The system should allow the user to land on to the homepage of BreachEscape when the customer enters a valid URL.
- The system should provide the login page to the user.
- For a new user, system asks the customer to fill in details like First name, Last name, Email id, Password, company name, designation and reason to use risk calculator.
- The system should allow the user to enter login details i.e., username and password if he is already registered.
- The system should allow the user to login to their account
- The system should display the correct error message in case of failure to login ex: invalid username/ password.
- The system should allow the user to go back to BreachEscape's homepage.

3. Business Process 3: Registered customer has forgotten his password

- System should allow the user to enter his login details.
- If the user has forgotten his password, system should allow the user to click on a forgot password hyperlink on the login page.

- System should notify the customer with a new password in email.
- System should allow the user to enter the new password and let the user login to their account with the new password.
- System should notify the user with an error message if the new password entered is incorrect.

4. **Business Process 4: Customer clicks on the Risk calculator on the navigation menu bar on homepage**

- The system should navigate user to the Assessment calculator page.
- If the user is not logged in to the BreachEscape website, system should allow the user to login and view the content on Risk assessment calculator page.
- If the user is already logged in, system should display a form with 19 fields and a submit button.
- The system should allow user to select options from the dropdowns and submit their answers.
- The system should validate that the user has filled out the entire questionnaire.
- The system should calculate the Risk Index based on the user inputs.
- The system should allow the users to view the calculated Risk Index.
- The system should display a gauge marking the risk based on the risk index.
- The system should also show score description based on the risk index.
- The system should allow the user to go back to their home page.

5. **Business Process 5.1: Customer clicks on the filters provided on the first interactive dashboard to gain specific insights**

- The system should navigate the user to the interactive dashboard page.
- System should allow the user to view the content on Interactive dashboards page for better understanding of breaches happening around the world.
- The dashboards will be dynamic which will reflect visualizations as per user selected filters.
- The system should allow the user to view States with breach occurrences and number of records lost based on selected years.
- The system will allow user to view filtered analytical reports.
- The system would show generic analysis of all the data breaches to data.
- The system would be able to represent filtered analysis based on different parameters.
- The system should allow the user to go back to their home page.

5. **Business Process 5.2: Customer clicks on the filters provided on the second interactive dashboard to gain specific insights**

- The system should navigate the user to the interactive dashboard page.
- System should display a link/button to view the interactive dashboard showing trend based on the historical data.

- The dashboards will be dynamic which will reflect visualizations as per user selected filters.
- The system should allow the user to view type of breach and number of breach by selecting the year of breach
- The system will allow user to view filtered analytical reports.
- The system would show generic analysis of all the data breaches to data.
- The system would be able to represent filtered analysis based on different parameters.
- The system should allow the user to go back to their home page.

6. Business Process 6: Customer wants to logout of the system

- The system should provide the user with an option to log out.
- The system should allow the user to log out.
- The system should successfully log out the user from their account and take them back to BreachEscapes's homepage.

10.1.2 Information oriented:

- The system should contain survey data from the companies recording their responses for the survey questions.
- The system should contain historical data for the occurrence of breaches.

10.2 Non-Functional requirements:

10.2.1 Operational

- The system should be a web-based application.
- The system should be cross browser compatible.

10.1.3 Security

- The system should keep all the data used for prediction private and confidential
- The system should keep all the data entered by user private and confidential

10.1.4 Performance

- The system should be up and running round the clock
- The system should be able to provide calculated results in less than 5 seconds
- The system should support at least 300 users at any given time

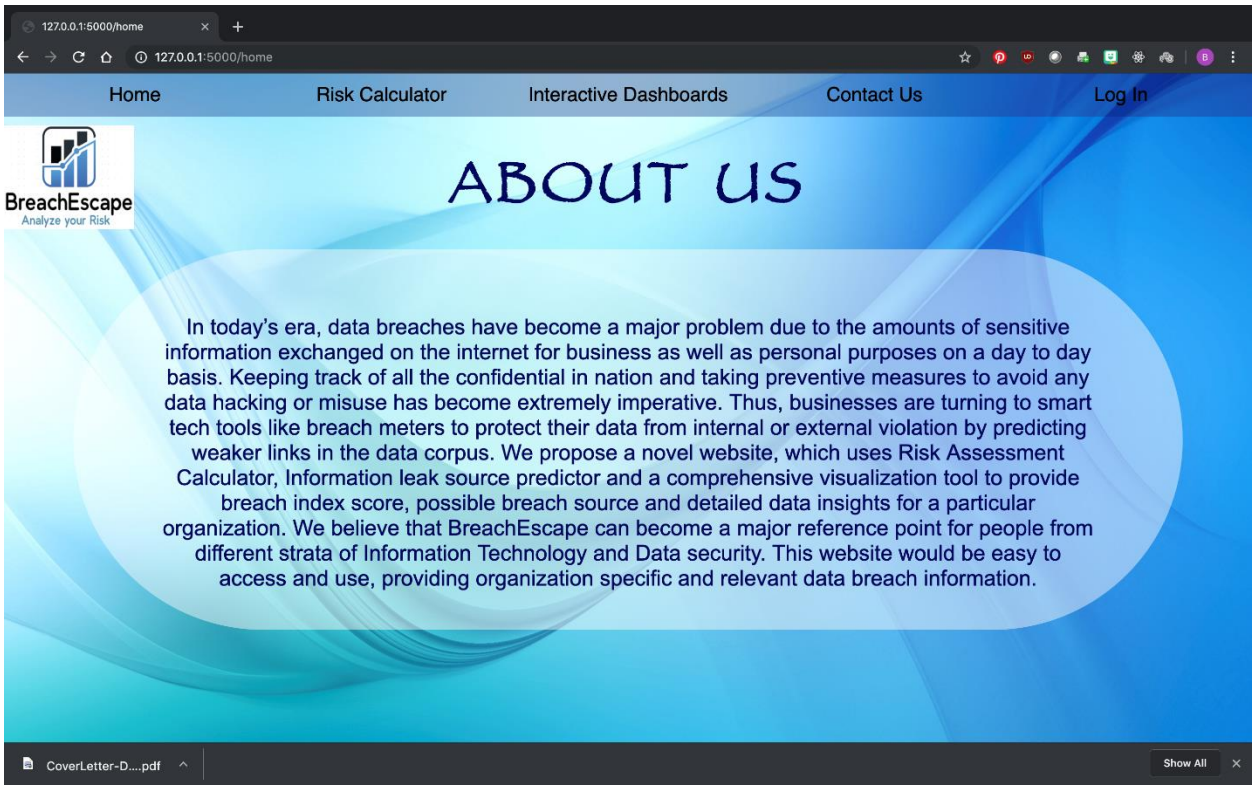
10.1.5 Cultural

- The system should not be biased among organization based on their location (Country/region)
- The system should not be biased among organization based on their type

11. Use Cases

Use Case 1

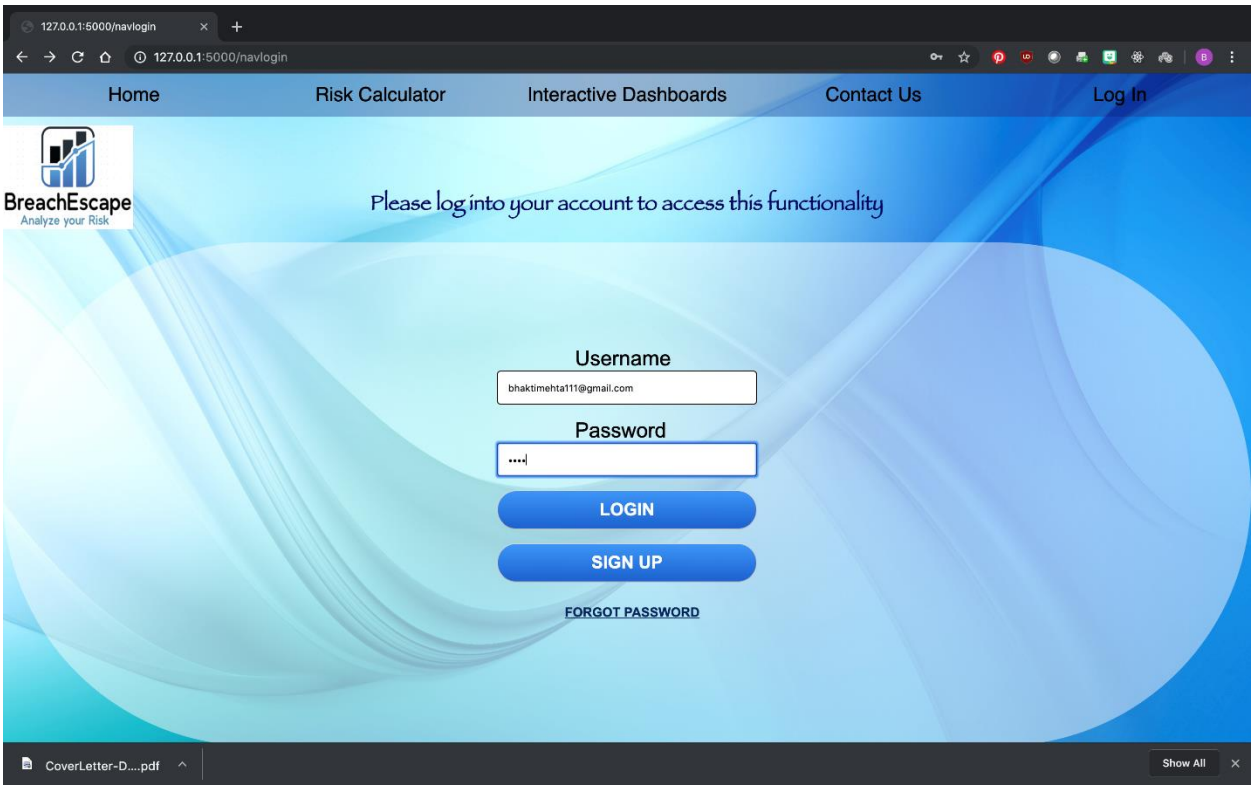
<table>
<tr><td colspan="2">Use Case Name: Customer lands on the homepage of BreachEscape’s Website.</td><td>Use case ID: 1</td><td>Importance level: High</td></tr>
<tr><td colspan="4">Primary Actor: 1. User/Customer/Company official/Security Solution provider
2. System/Admin</td></tr>
<tr><td colspan="4">Short Description: This describes how a customer lands on the homepage of BreachEscape and is able to navigate through the menu and its options</td></tr>
<tr><td colspan="4">Trigger: User wants to view all the options they get on BreachEscape’s website
Type: External</td></tr>
<tr><td colspan="4">Preconditions: User knows the URL for the website.
Has an updated browser to display the website.</td></tr>
<tr><td colspan="2">Steps Performed:
1. User enters the URL into the browser and hits enter.
2. User lands on the BreachEscape website and lands on Homepage.
3. User can see a navigation menu bar with Home, Contact Us, Risk calculator, Interactive dashboard and Login options.
4. User clicks on all the menu items one by one and navigates to respective pages.</td><td colspan="2">Information:
Required URL and compatible browser</td></tr>
<tr><td colspan="4">Post Condition: User could address his accessibility to all the menu items and their respective webpages.</td></tr>
<tr><td colspan="2">Major Inputs:
URL
Click on Menu items</td><td colspan="2">Major outputs
Description
Navigation through items
Page availability</td></tr>
</table>



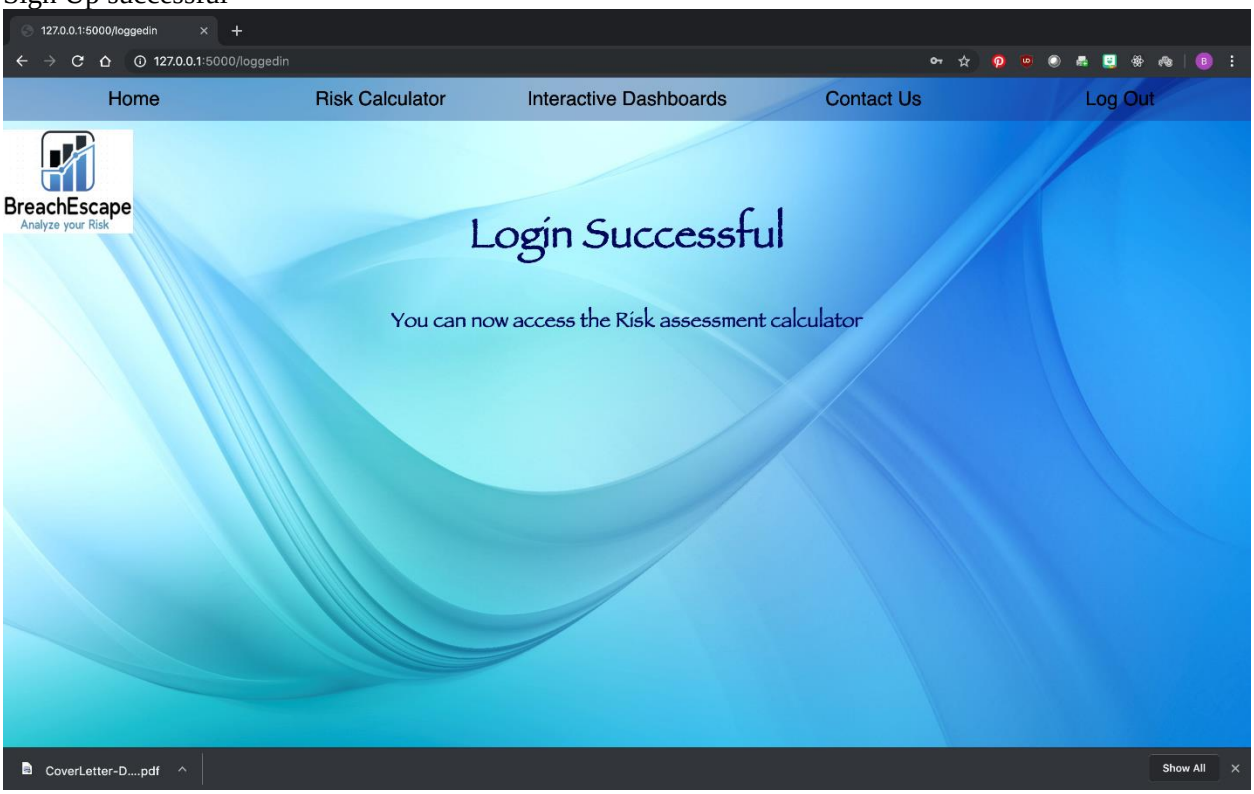
Use Case 2

Use Case Name: User/Customer Signs in to the BreachEscape Website	**Use case ID: 2**	*Importance: High*
Primary Actor: 1. User/Customer/Company official/Security Solution provider 2. System/Admin		
Short Description: This describes how a user lands on the BreachEscape website and Signs in/up on the website		
Trigger: User wants to view the services provided by BreachEscape that are restricted to members only **Type:** External		
Preconditions: User knows the URL for the website. Has an updated browser to display the website. User has already registered and has valid credentials.		

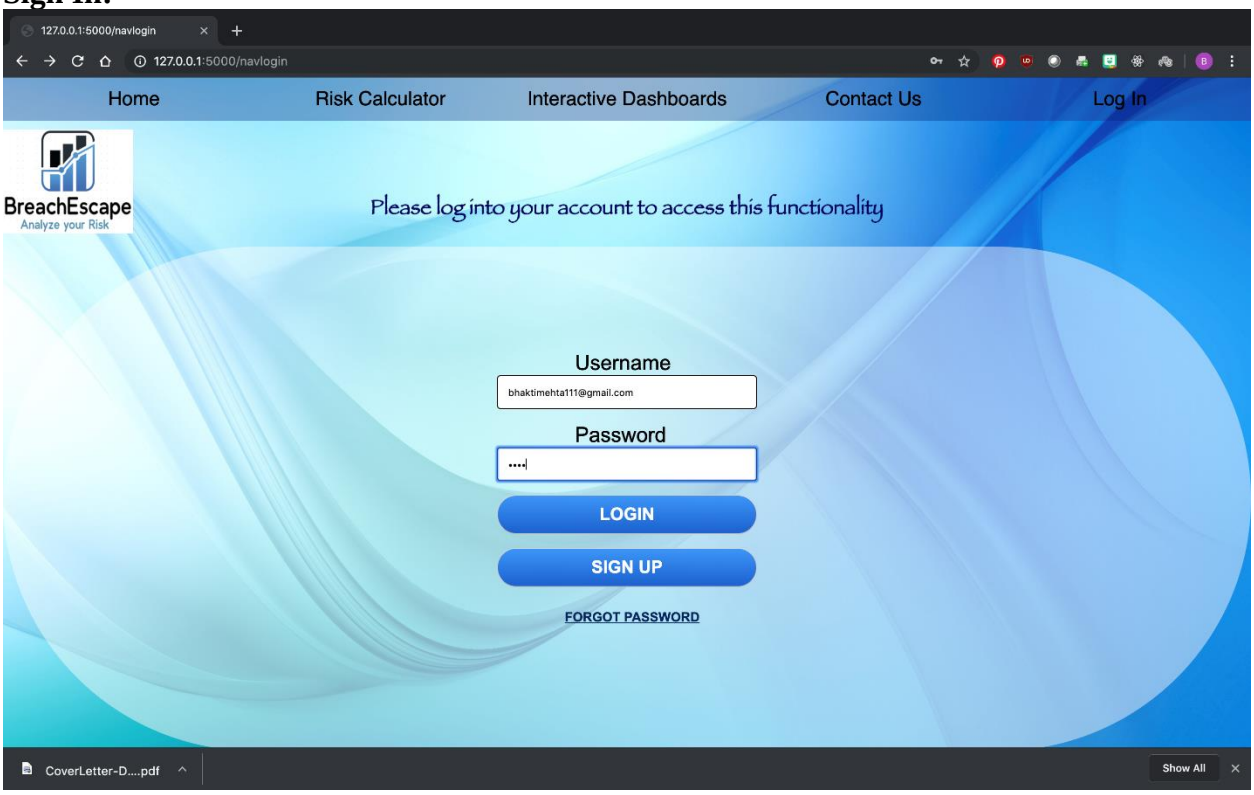
Steps Performed: Normal Course1: 1. User enters the URL into the browser and hits enter. 2. User lands on the BreachEscape website and clicks on Homepage. 3. User Clicks on Login option on the menu bar. 4. User enters Username and password and clicks on Sign in 5. Credentials if valid user can see"Login is successful"	**Information:** Required URL and compatible browser Username and Password
Post Condition: User has signed in to their account with BreachEscape.	
Normal Course 2: From step 3 Normal course 1 4. User clicks on Sign up button 5. User enters his details in the signup form clicks on submit button 6. User can see a notification saying "Thank you" 6.User enters Username and password and clicks on Sign in 7.Credentials if valid user can see"Login is successful" **Exception: Step 4** 4. User enter invalid credentials. 5. User is notified with an error "Incorrect username or password Please try again to login".	
Major Inputs: URL Click on Menu items Username password	**Major outputs** Navigation through items Successful Login Error notification

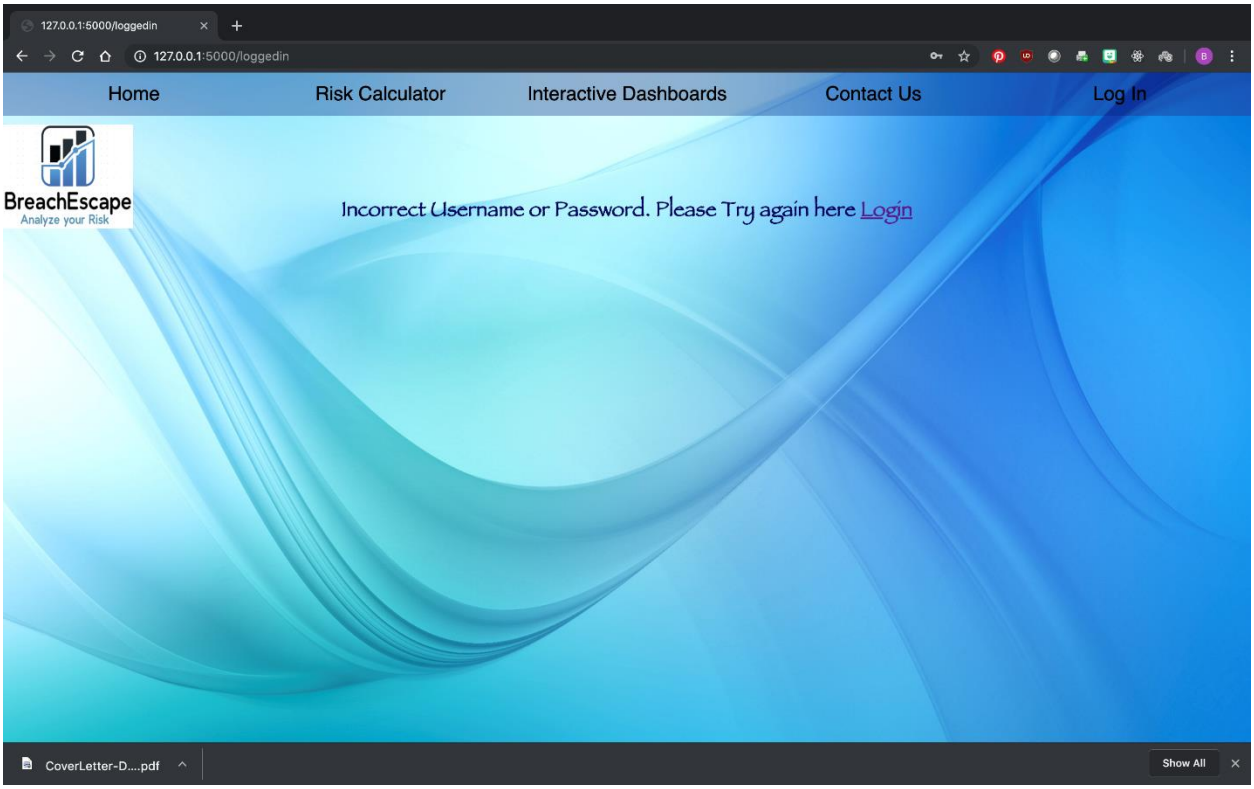


Sign Up successful

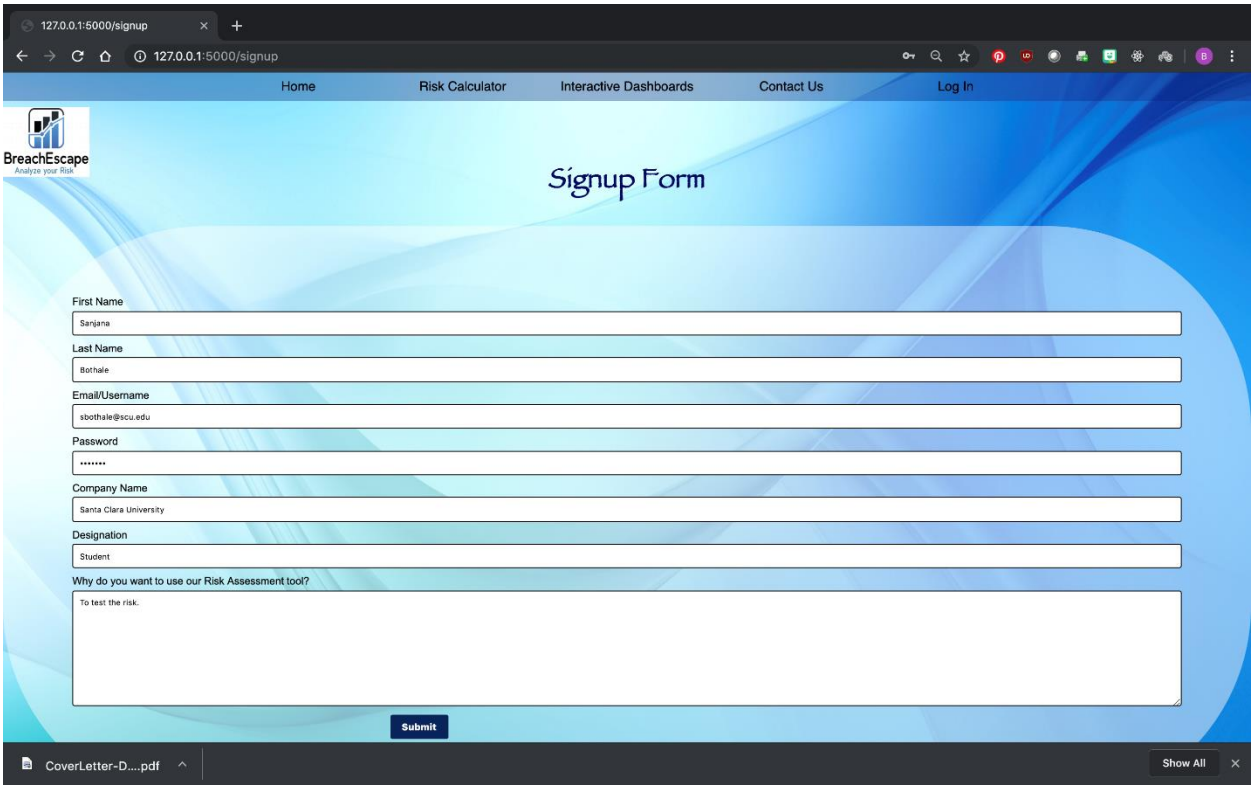


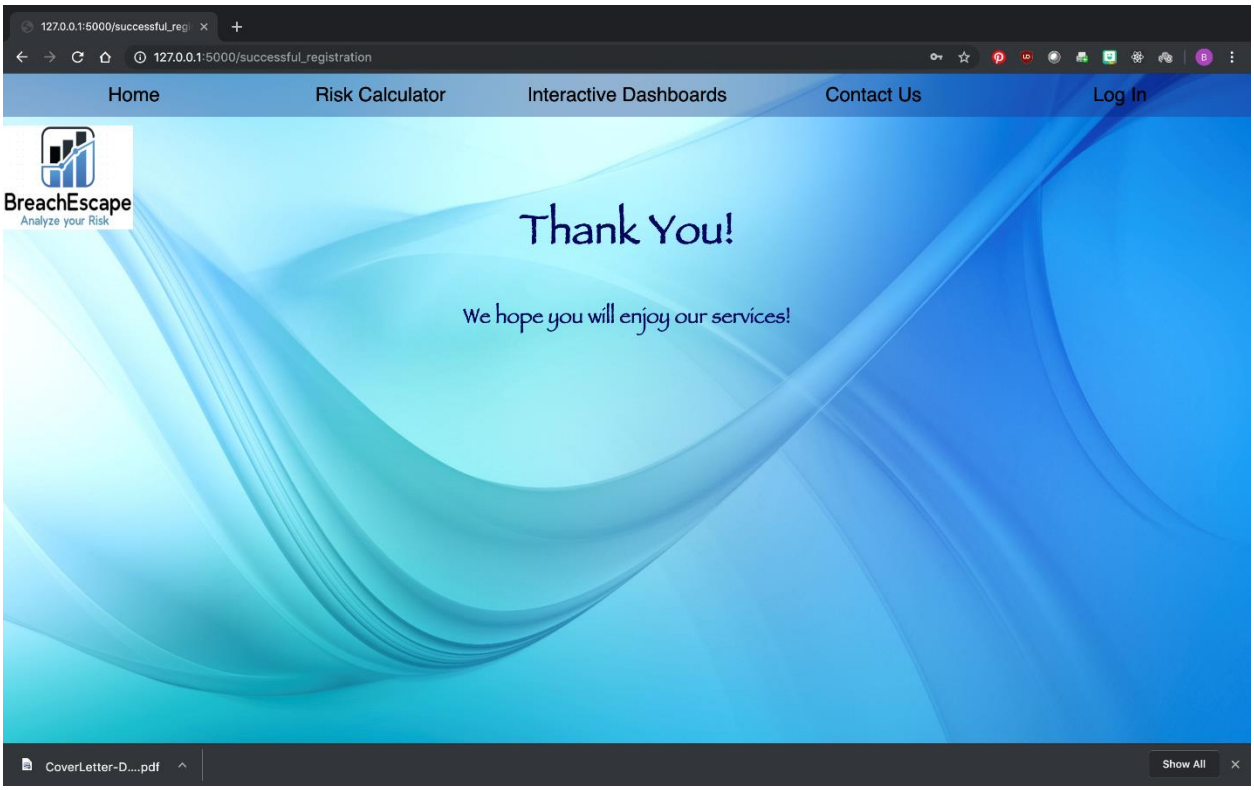
Sign In:


127.0.0.1:5000/navlogin
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Please log into your account to access this functionality
Username
bhaktimehta111@gmail.com
Password
LOGIN
SIGN UP
FORGOT PASSWORD
CoverLetter-D....pdf
Show All


127.0.0.1:5000/loggedin
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Incorrect Username or Password. Please Try again here Login
CoverLetter-D....pdf
Show All

Sign up:


127.0.0.1:5000/signup
127.0.0.1:5000/signup
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Signup Form
First Name
Sanjana
Last Name
Bothale
Email/Username
sbothale@scu.edu
Password
Company Name
Santa Clara University
Designation
Student
Why do you want to use our Risk Assessment tool?
To test the risk.
Submit
CoverLetter-D....pdf
Show All


127.0.0.1:5000/successful_reg
127.0.0.1:5000/successful_registration
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Thank You!
We hope you will enjoy our services!
CoverLetter-D....pdf
Show All

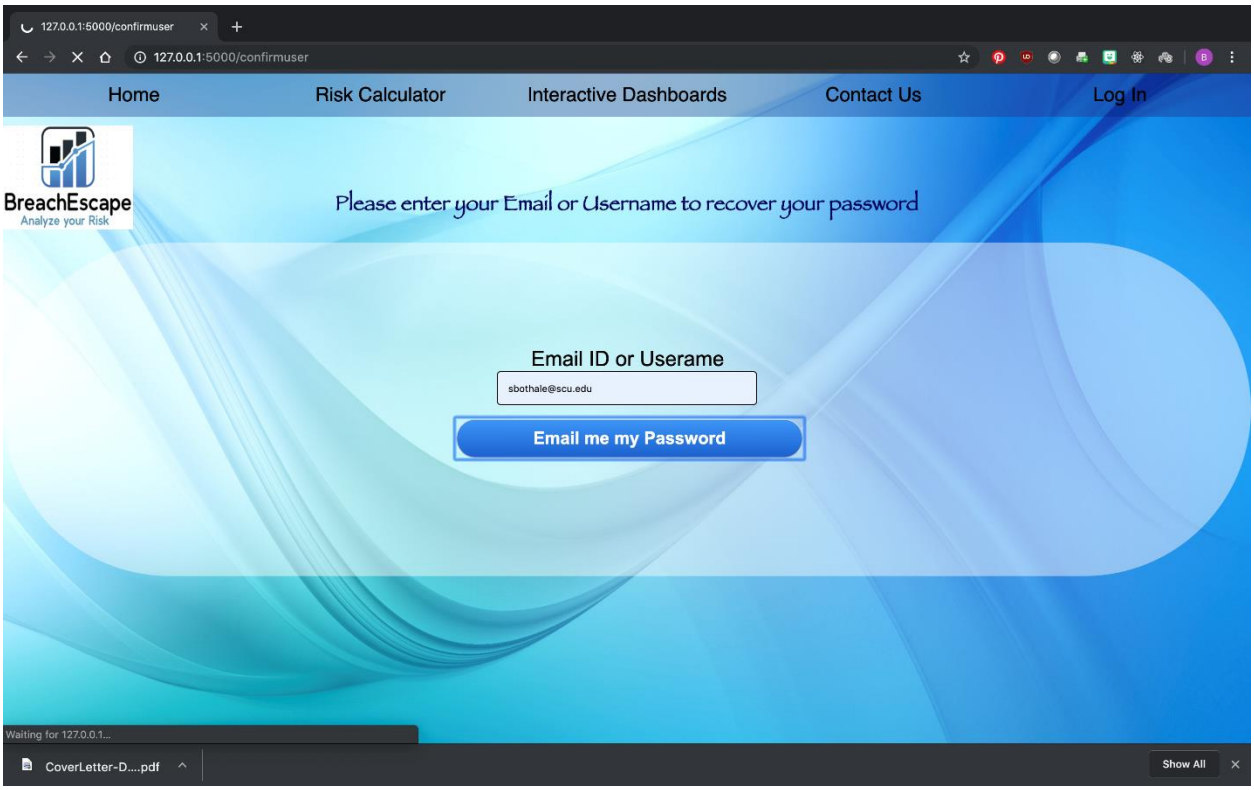
Use Case 3

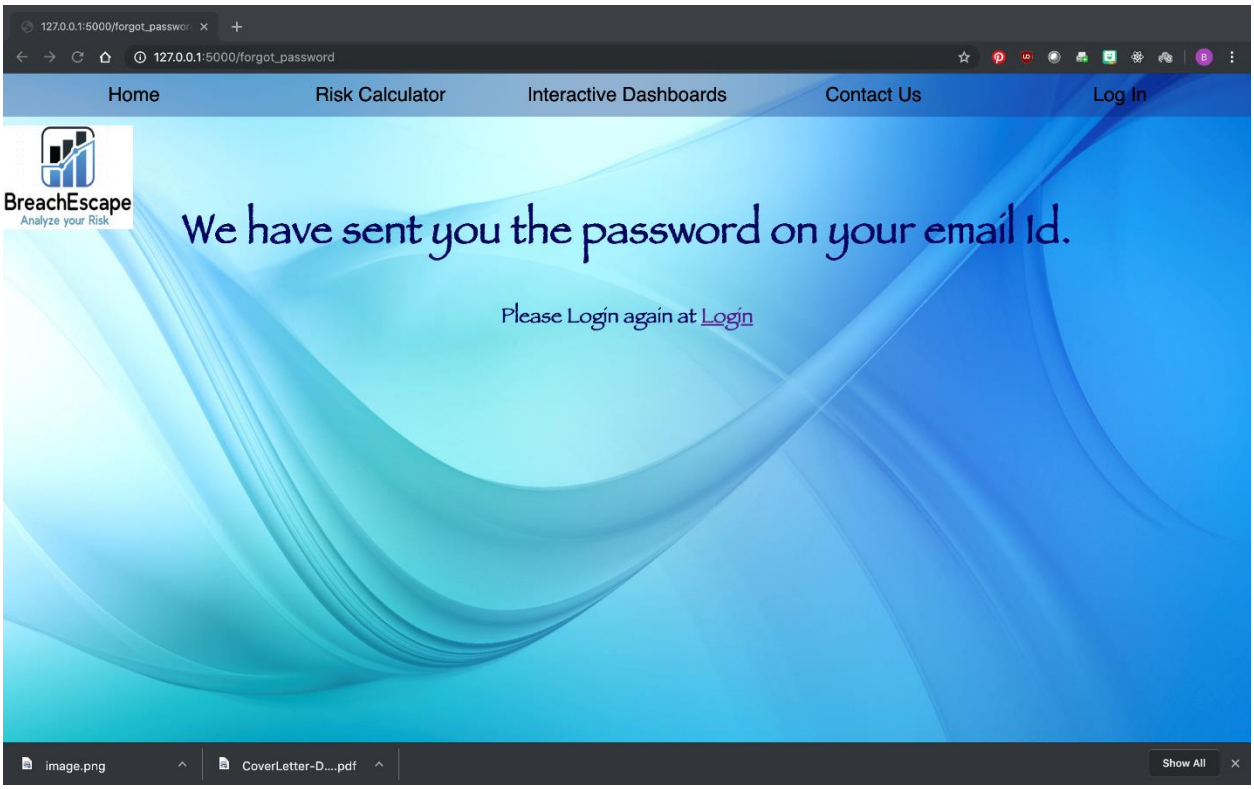
<table>
<tr><td colspan="2">Use Case Name: Customer/user has forgotten his password and wants to login to his BreachEscape’s account</td><td>Use case ID: 3</td><td>Importance: High</td></tr>
<tr><td colspan="4">Primary Actor: 1. User/Customer/Company official/Security Solution provider
2. System/Admin</td></tr>
<tr><td colspan="4">Short Description: This describes how customer on forgetting his password requests for the password to Sign in to their account</td></tr>
<tr><td colspan="4">Trigger: User wants to login to their BreachEscape’s website
Type: External</td></tr>
<tr><td colspan="4">Preconditions: User knows the URL for the website.
User knows the Username</td></tr>
<tr><td colspan="2">Steps Performed:
1. User enters the URL into the browser and hits enter.
2. User lands on the BreachEscape website and clicks on Homepage.
3. User navigates to the Login page
4. Clicks on “Forgot password”
5. System asks for the valid email id and User clicks on Email me my password.
6. System sends the password to the registered email id.
7. System shows “We have sent you the password on your email id….Please Login again at <u>LOGIN</u>”</td><td colspan="2">Information:
Required URL and compatible browser

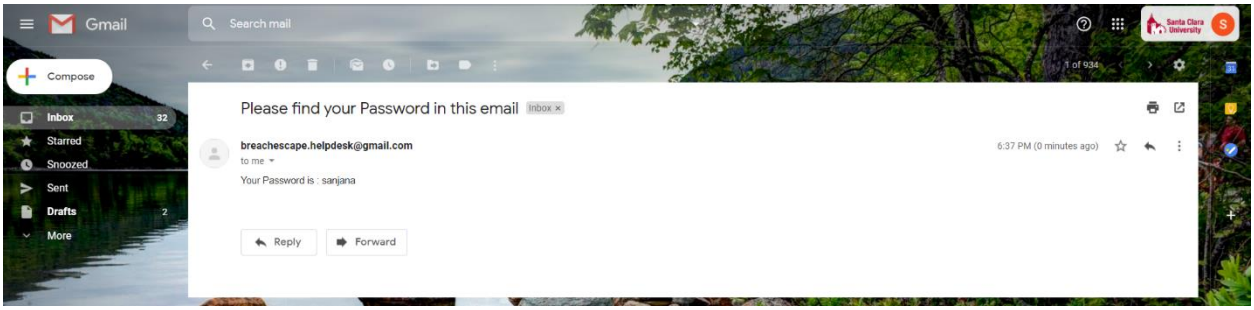
Email id
Emailed password

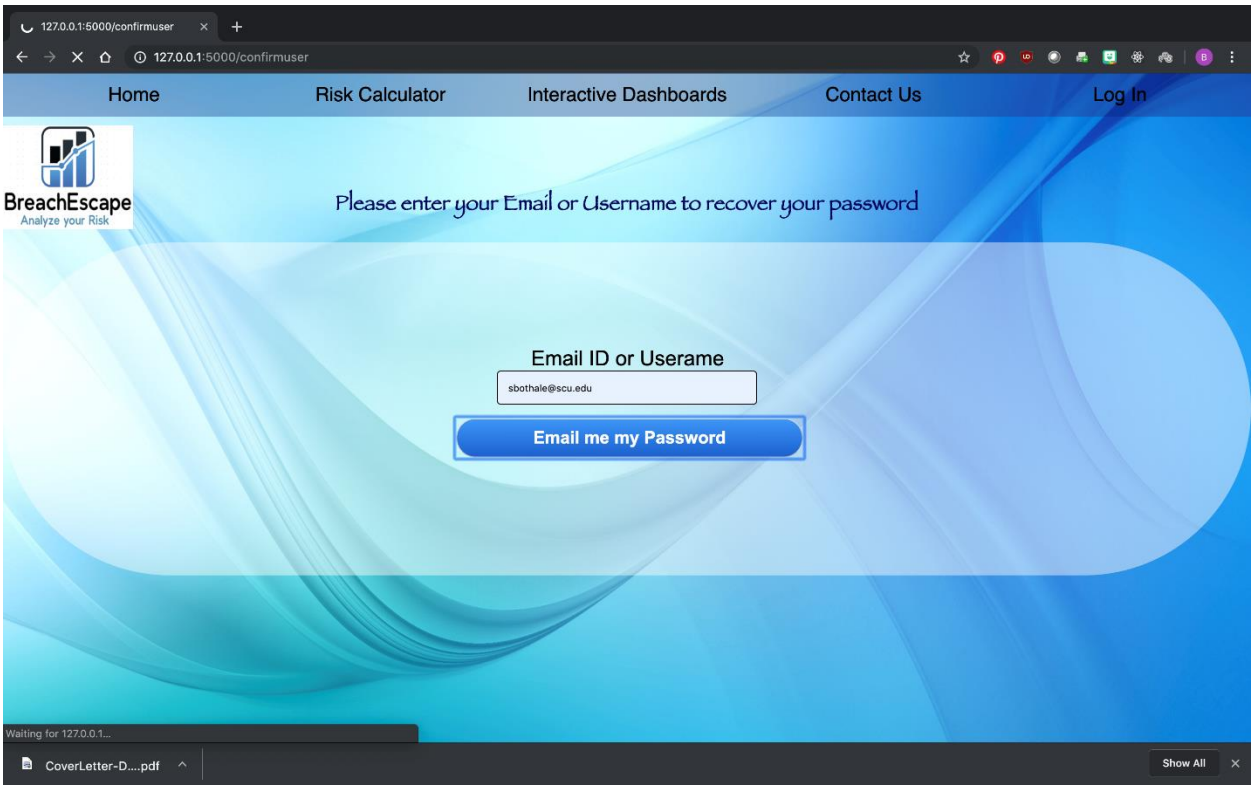
Notification</td></tr>
<tr><td colspan="4">Post Condition: User Login is successful</td></tr>
<tr><td colspan="4">Exception:Step 5
5. User enters an email address that is not registered with Breachescape and User clicks on Email me my password.
6. User is notified that “We cannot find you in our database”.
7.System provides user with an option to enter his email id again.</td></tr>
</table>

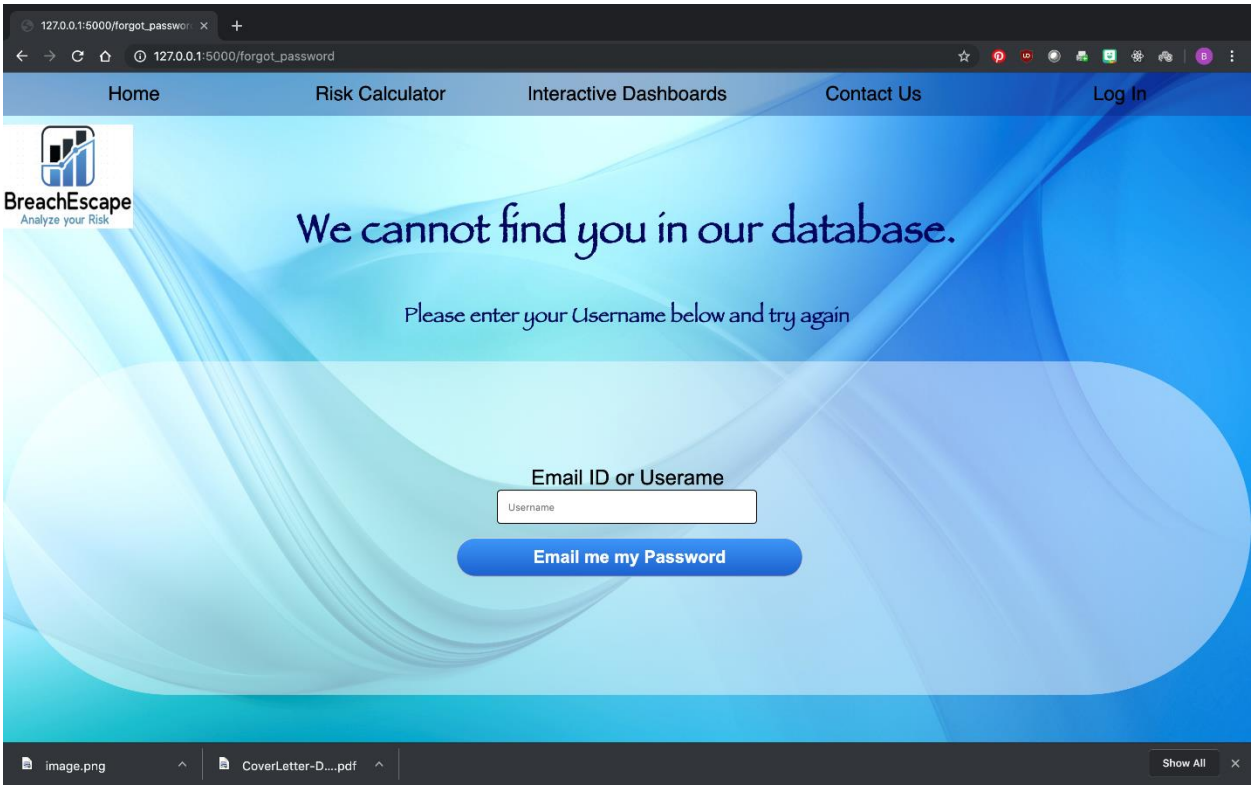
Major Inputs: URL Username password Email id	**Major outputs**: Email with password Successful login Error message




127.0.0.1:5000/forgot_password
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
We have sent you the password on your email Id.
Please Login again at Login
image.png
CoverLetter-D....pdf
Show All


Gmail
Search mail
Compose
Inbox 32
Starred
Snoozed
Sent
Drafts 2
More
Please find your Password in this email
breachescape.helpdesk@gmail.com
to me
6:37 PM (0 minutes ago)
Your Password is : sanjana
Reply
Forward


127.0.0.1:5000/confirmuser
127.0.0.1:5000/confirmuser
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Please enter your Email or Username to recover your password
Email ID or Userame
sbothale@scu.edu
Email me my Password
Waiting for 127.0.0.1...
CoverLetter-D....pdf
Show All

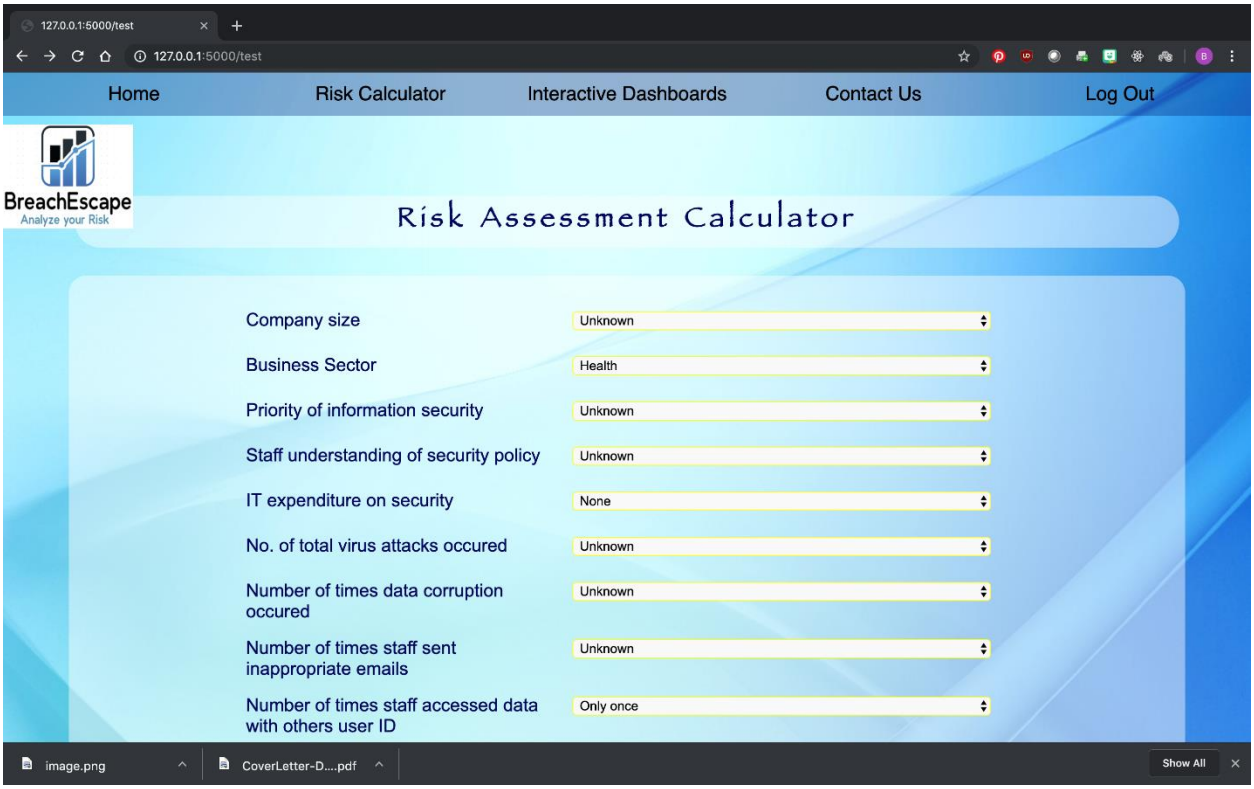

127.0.0.1:5000/forgot_passwor
127.0.0.1:5000/forgot_password
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
We cannot find you in our database.
Please enter your Username below and try again
Email ID or Userame
Username
Email me my Password
image.png
CoverLetter-D....pdf
Show All

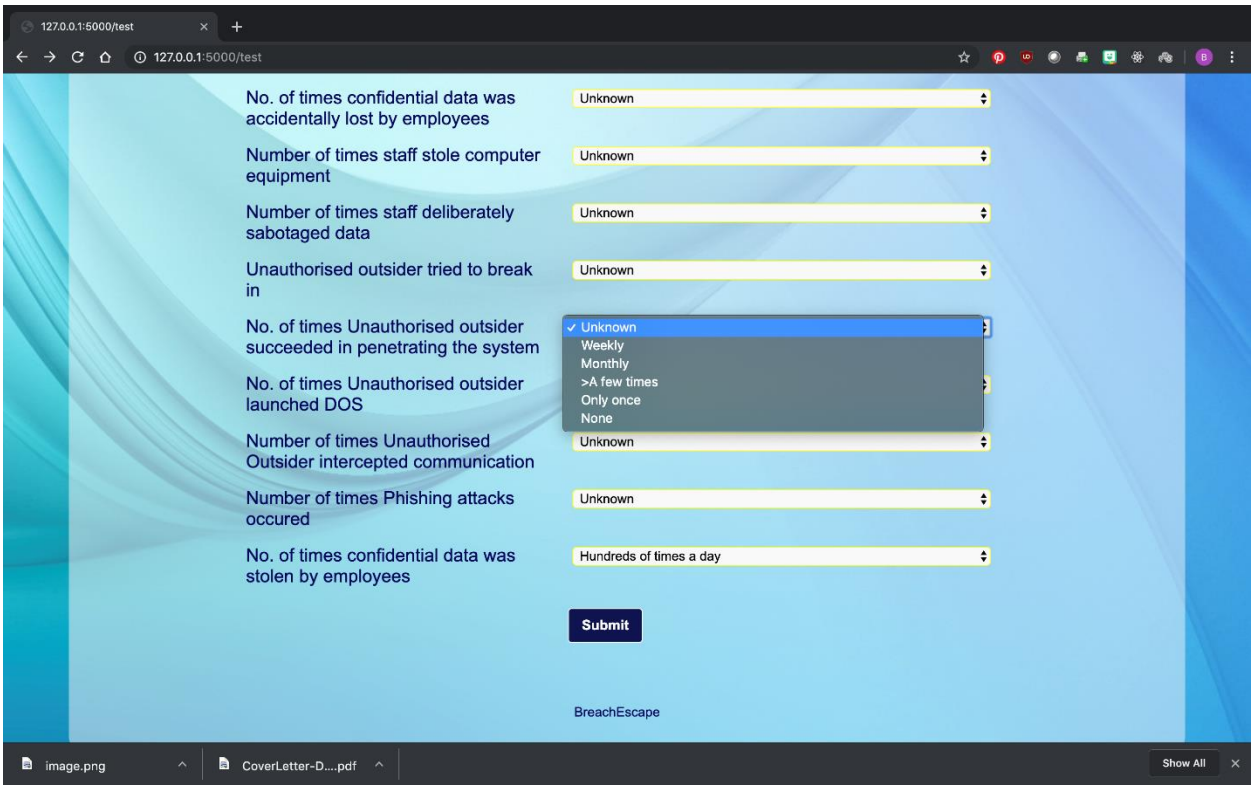
Use Case 4

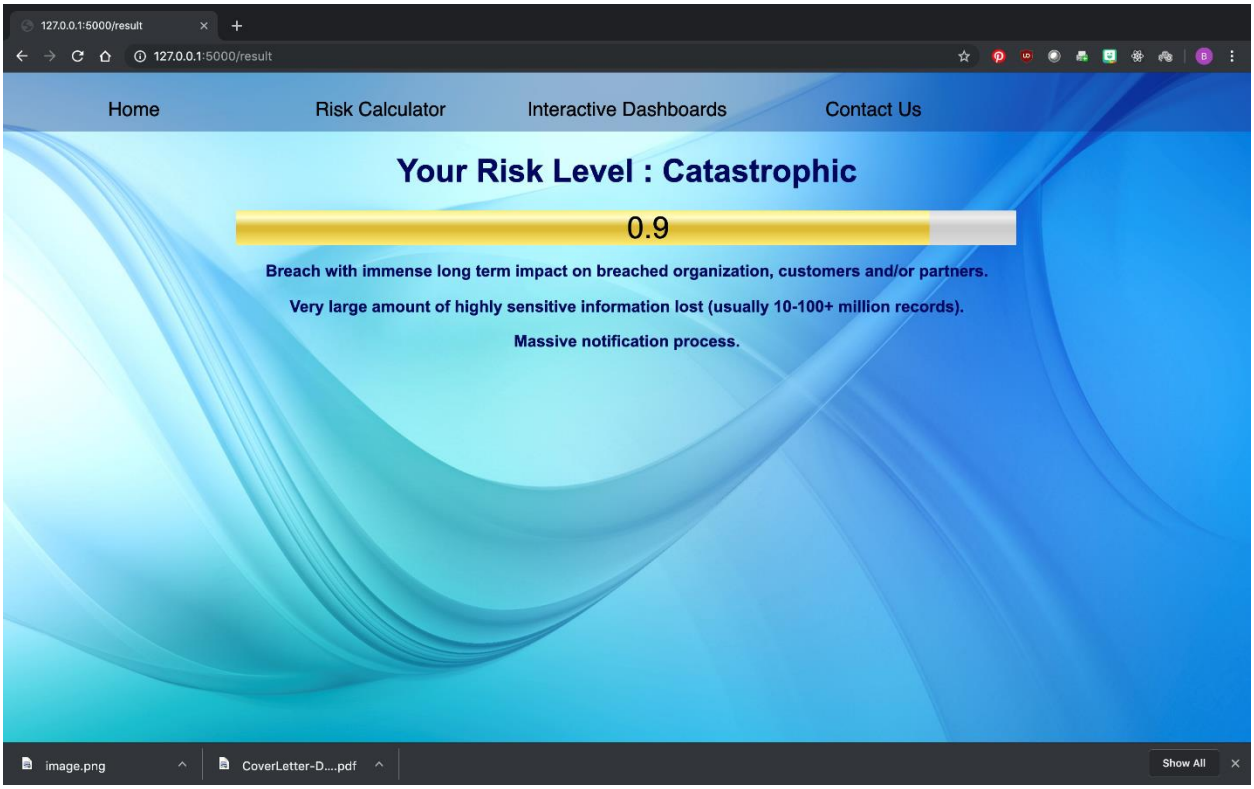
<table>
<tr><td colspan="2">Use Case Name: Customer clicks on the Risk assessment calculator on the navigation menu bar.</td><td>Use case ID: 4</td><td>Importance level: High</td></tr>
<tr><td colspan="4">Primary Actor: 1. User/Customer/Company official/Security Solution provider
2. System/Admin</td></tr>
<tr><td colspan="4">Short Description: Customer wants to check their risk score based on some parameters and fills in the form for the same</td></tr>
<tr><td colspan="4">Trigger: User wants to check their susceptibility to Breach
Type: External</td></tr>
<tr><td colspan="4">Preconditions: User is already logged-in</td></tr>
<tr><td colspan="2">Steps Performed: Normal Course 1-
1. User Clicks on the Risk assessment calculator option on the menu bar.
2. User is navigated to the Risk assessment calculator page.
3. User can see 19 questions with drop down fields to select answers from.
4. User selects options for each field from the drop down menu.
5. User clicks on the submit button to submit his responses.
6. A risk score, gauge and Risk score description are displayed</td><td colspan="2">Information:
Company details
Answers to survey questions

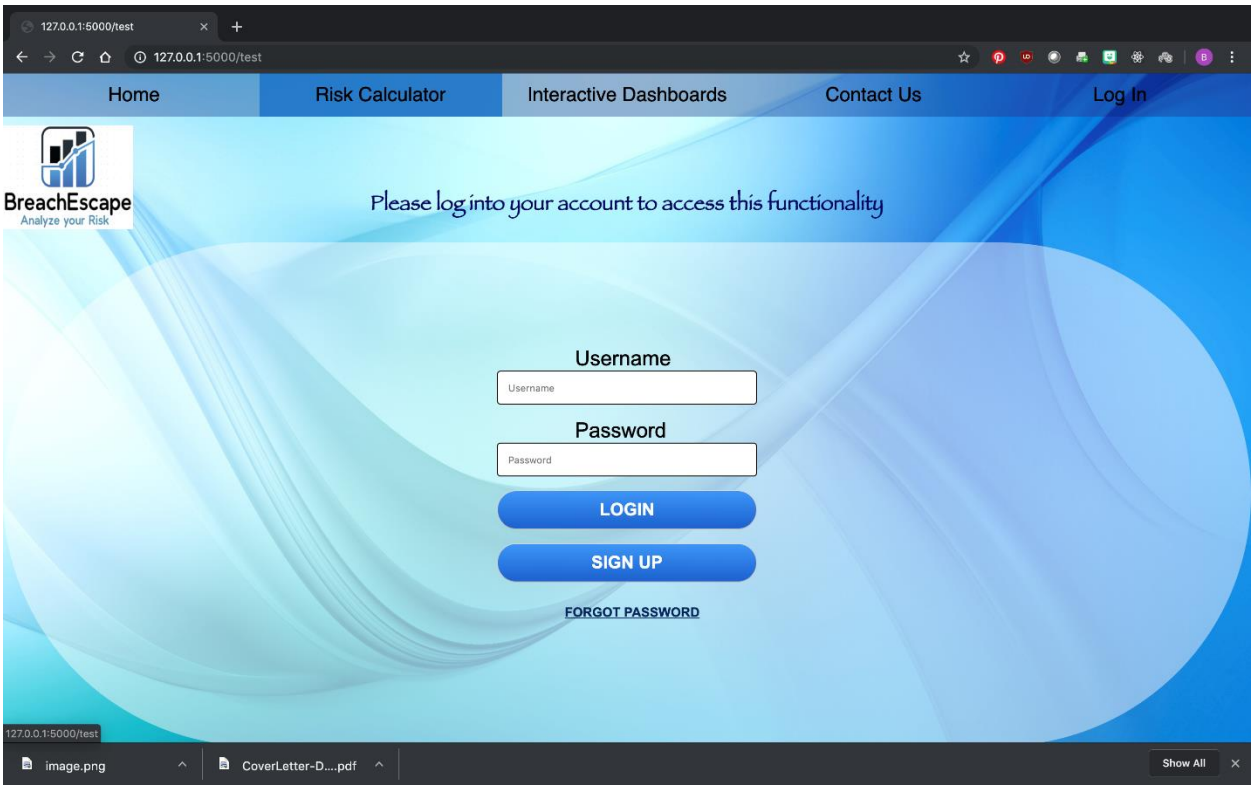
Form submission</td></tr>
<tr><td colspan="4">Post Condition: User can view if their company is at risk based on the risk score and could take action based on the recommendations.</td></tr>
<tr><td colspan="4">Normal Course 2: After normal course 1-Step 2:
2. User is not signed-in and navigates to the Risk assessment calculator page.
3. User enters their valid credentials and logs in to their account.
4. Normal Course 1-Step 3 continued.
Exception: After Normal course 1-Step 4
4. User does not select answer for a question.
5. System considers a default value for the unanswered question.
6. Normal Course 1-step 5 continued.</td></tr>
</table>

Major Inputs: Company details Survey answers Username, password	**Major outputs** Survey questions Risk score Score description




127.0.0.1:5000/test
No. of times confidential data was accidentally lost by employees
Unknown
Number of times staff stole computer equipment
Unknown
Number of times staff deliberately sabotaged data
Unknown
Unauthorised outsider tried to break in
Unknown
No. of times Unauthorised outsider succeeded in penetrating the system
Unknown
Weekly
Monthly
>A few times
Only once
None
No. of times Unauthorised outsider launched DOS
Number of times Unauthorised Outsider intercepted communication
Unknown
Number of times Phishing attacks occured
Unknown
No. of times confidential data was stolen by employees
Hundreds of times a day
Submit
BreachEscape
image.png
CoverLetter-D....pdf
Show All

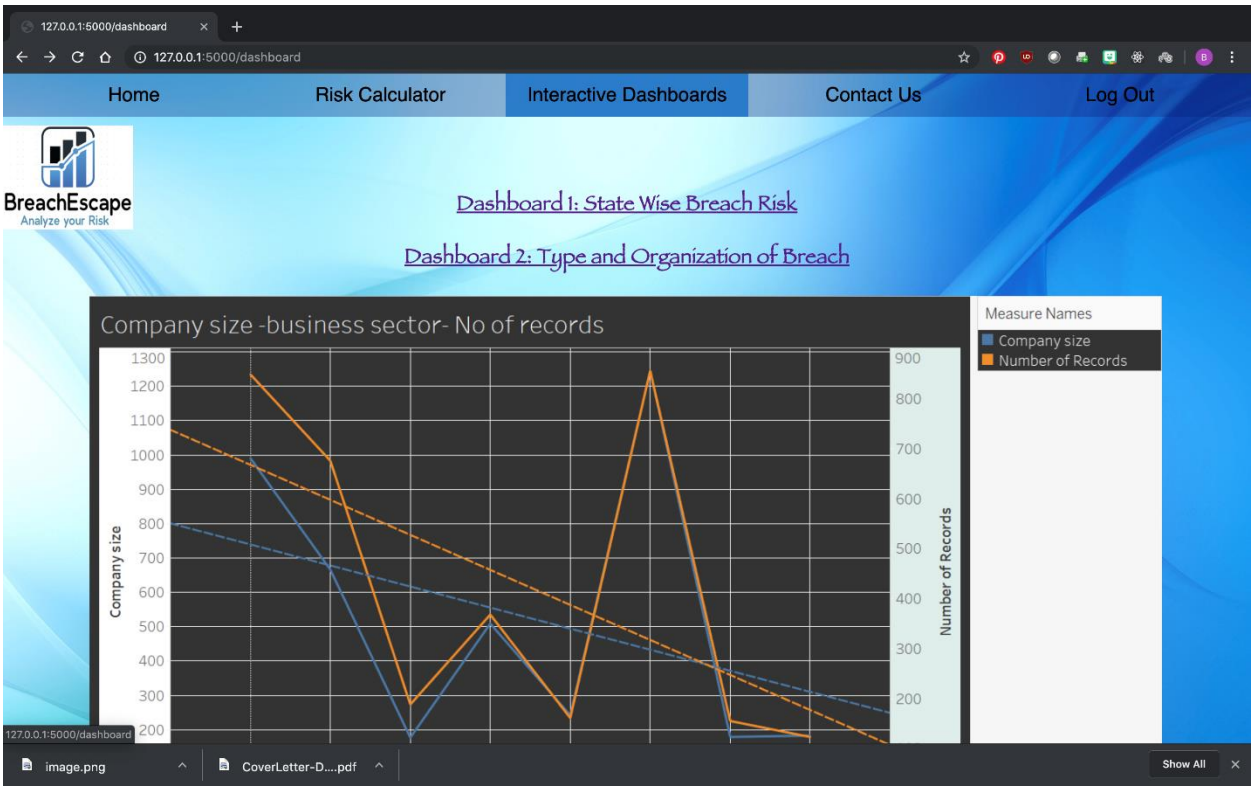

127.0.0.1:5000/result
Home
Risk Calculator
Interactive Dashboards
Contact Us
Your Risk Level : Catastrophic
0.9
Breach with immense long term impact on breached organization, customers and/or partners.
Very large amount of highly sensitive information lost (usually 10-100+ million records).
Massive notification process.
image.png
CoverLetter-D....pdf
Show All

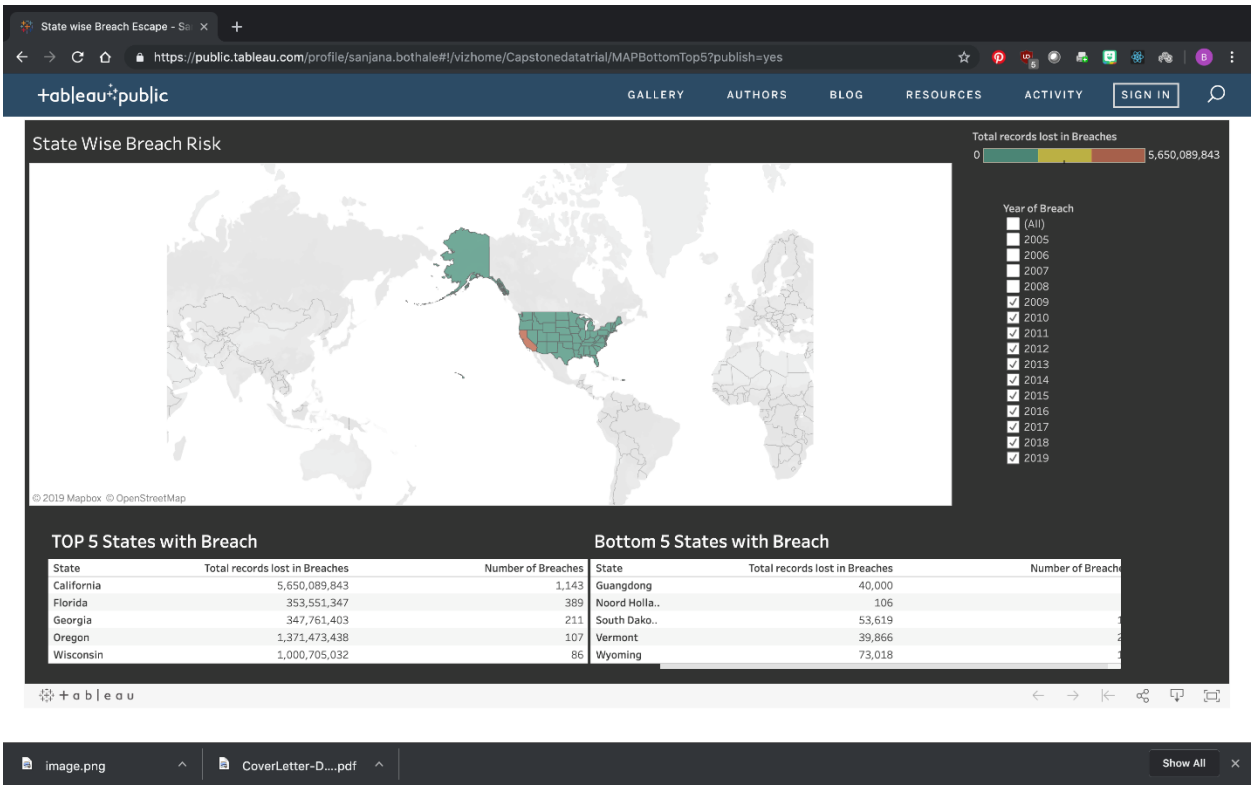


Use Case 5.1

Use Case Name: Customer clicks on the Interactive Dashboard on the navigation menu bar on Homepage and selects a dashboard to view	**Use case ID: 5.1**	*Importance level: High*
Primary Actor: 1. User/Customer/Company official/Security Solution provider 2. System/Admin		
Short Description: User can view trend based on type of breach, state of breach occurrence, year of breach and organization of breach		
Trigger: User wants to check the trend based on the historical occurrences of Breach **Type:** External		
Preconditions: User lands on the homepage and could see Interactive Dashboard on the menu bar.		
Steps Performed: 1. User enters the URL into the browser and hits enter. 2. User lands on the BreachEscape website and	**Information:** Required URL and compatible browser	

clicks on Homepage. 3. User clicks on Interactive Dashboards Menu. 4. The system should navigate the user to the interactive dashboard page. 5. User can see links with Statewise Breach risk and Type and organization of breach. 6. User clicks on State wise Breach Risk. 7. User selects Year of Breach and State from the filters. 8. Dashboard with Top 5 and bottom 5 results is seen	URL for the first dashboard Required Filters
Post Condition: User could address their accessibility to all the menu items and their respective webpages.	
Major Inputs: Website URL Click on Menu items Dashboard URLs Selection of first URL	**Major outputs** Description Navigation through items Page availability Interactive dashboard with trends

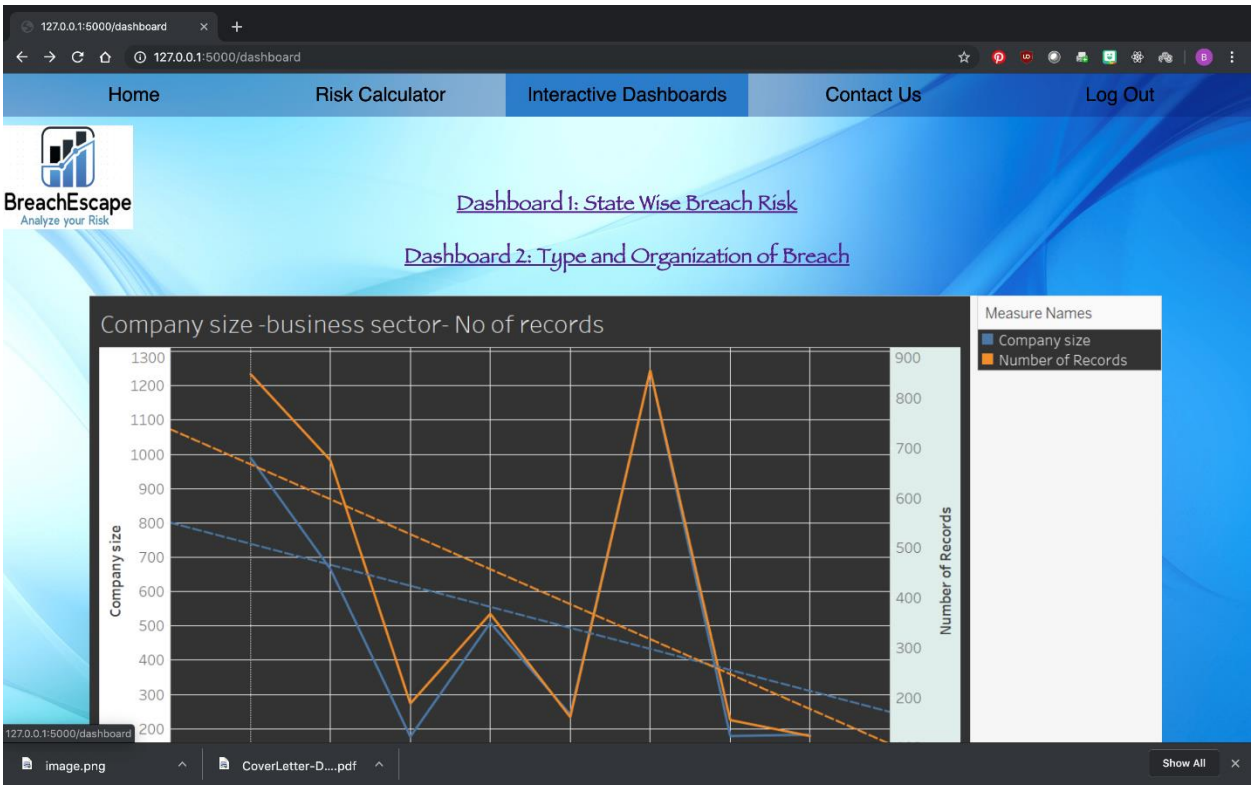


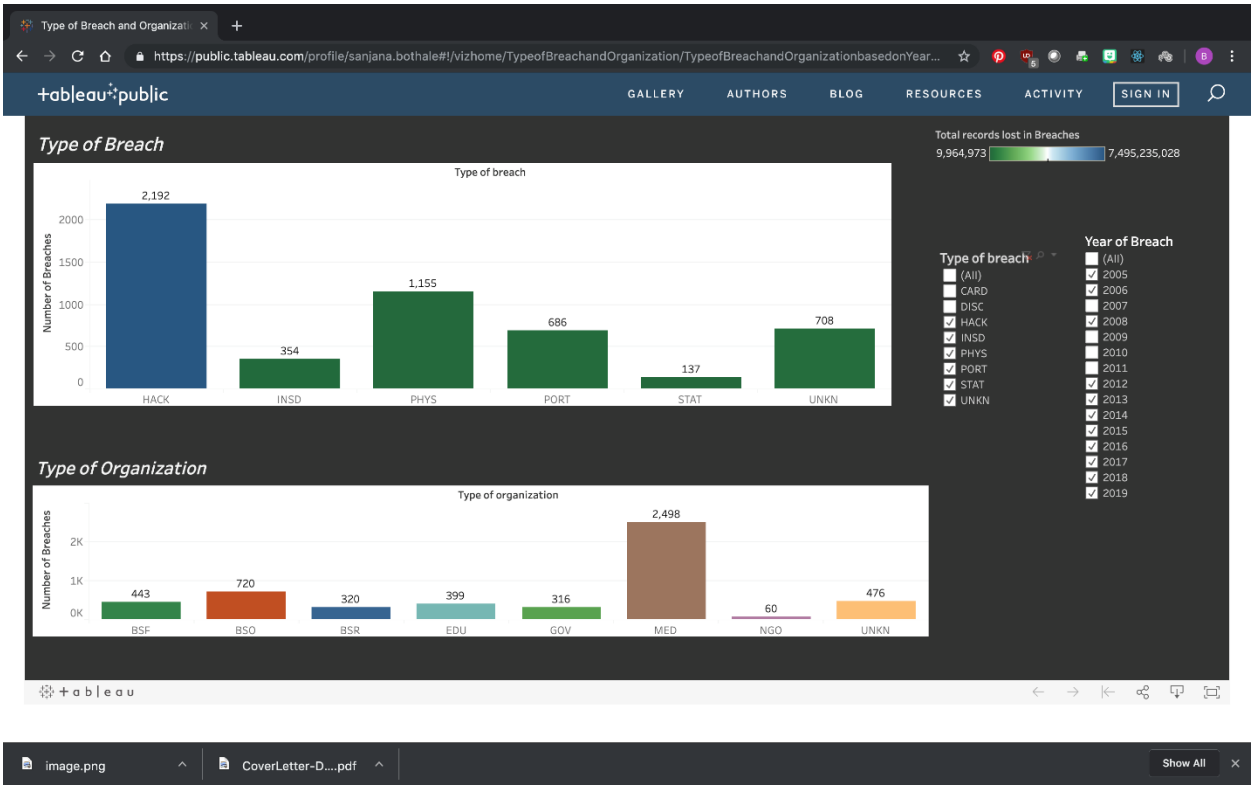


Use Case 5.2

Use Case Name: Customer clicks on the Interactive Dashboard on the navigation menu bar on Homepage and selects a dashboard to view	**Use case ID: 5.2**	*Importance level: High*
Primary Actor: 1. User/Customer/Company official/Security Solution provider 2. System/Admin		
Short Description: User can view trend based on Type of Breach and Type of Organization		
Trigger: User wants to check the trend based on the historical occurrences of Breach **Type:** External		

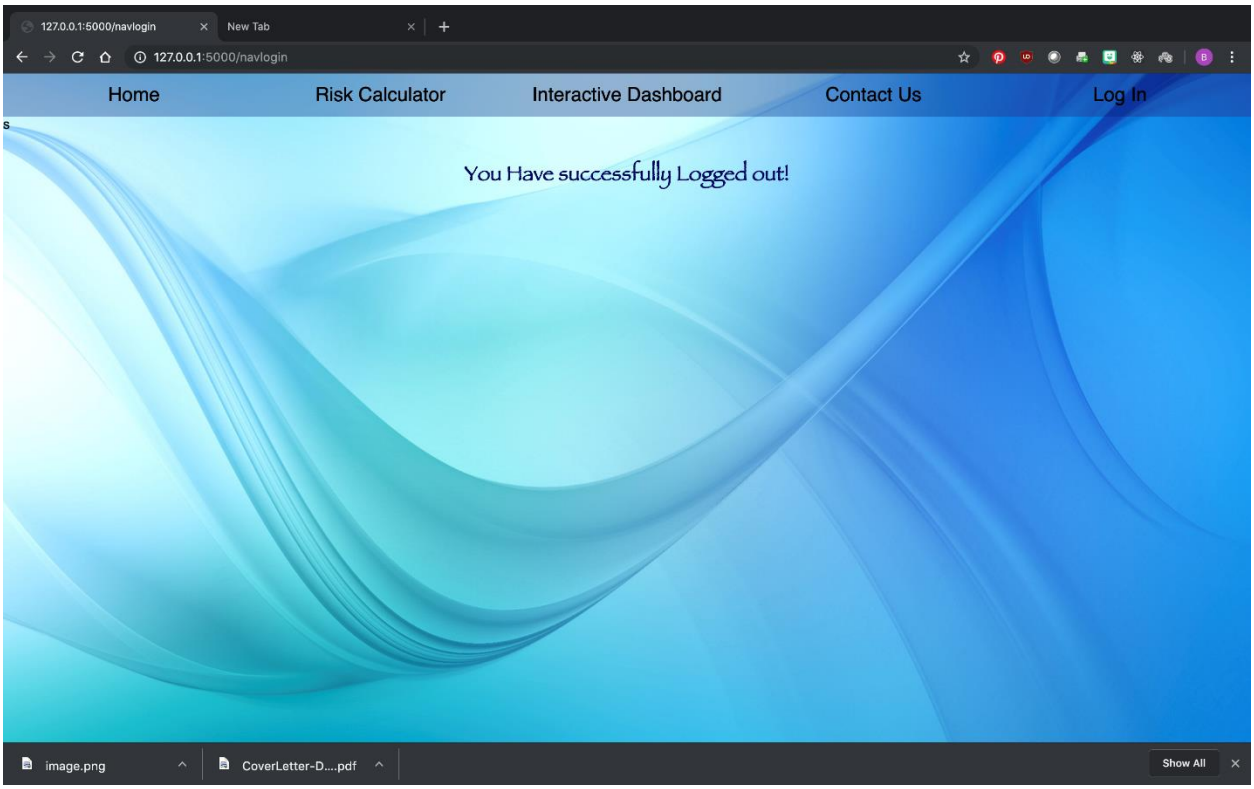
Preconditions: User lands on the homepage and could see Interactive Dashboard on the menu bar.	
Steps Performed: 1. User enters the URL into the browser and hits enter. 2. User lands on the BreachEscape website and clicks on Homepage. 5. User clicks on Interactive Dashboards Menu. 6. The system should navigate the user to the interactive dashboard page. 7. User can see links with Statewise Breach risk and Type and organization of breach. 8. User clicks on Type and Organization of breach. 9. User selects Year of Breach and Type of Breach from the filters. 10. Dashboard with Type of Breach and Type of Organization can be seen.	**Information:** Required URL and compatible browser URL for the first dashboard Required Filters
Post Condition: User could address their accessibility to all the menu items and their respective webpages.	
Major Inputs: Website URL Click on Menu items Dashboard URLs Selection of second URL on page	**Major outputs** Description Navigation through items Page availability Interactive dashboard with trends


127.0.0.1:5000/dashboard
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log Out
BreachEscape
Analyze your Risk
Dashboard 1: State Wise Breach Risk
Dashboard 2: Type and Organization of Breach
Company size -business sector- No of records
Company size
Number of Records
Measure Names
Company size
Number of Records
image.png
CoverLetter-D....pdf
Show All


Type of Breach and Organizati
https://public.tableau.com/profile/sanjana.bothale#!/vizhome/TypeofBreachandOrganization/TypeofBreachandOrganizationbasedonYear...
tableau public
GALLERY
AUTHORS
BLOG
RESOURCES
ACTIVITY
SIGN IN
Type of Breach
Type of breach
Number of Breaches
2,192
354
1,155
686
137
708
HACK
INSD
PHYS
PORT
STAT
UNKN
Total records lost in Breaches
9,964,973
7,495,235,028
Type of breach
(All)
CARD
DISC
HACK
INSD
PHYS
PORT
STAT
UNKN
Year of Breach
(All)
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
Type of Organization
Type of organization
Number of Breaches
443
720
320
399
316
2,498
60
476
BSF
BSO
BSR
EDU
GOV
MED
NGO
UNKN
image.png
CoverLetter-D....pdf
Show All

Use Case 6

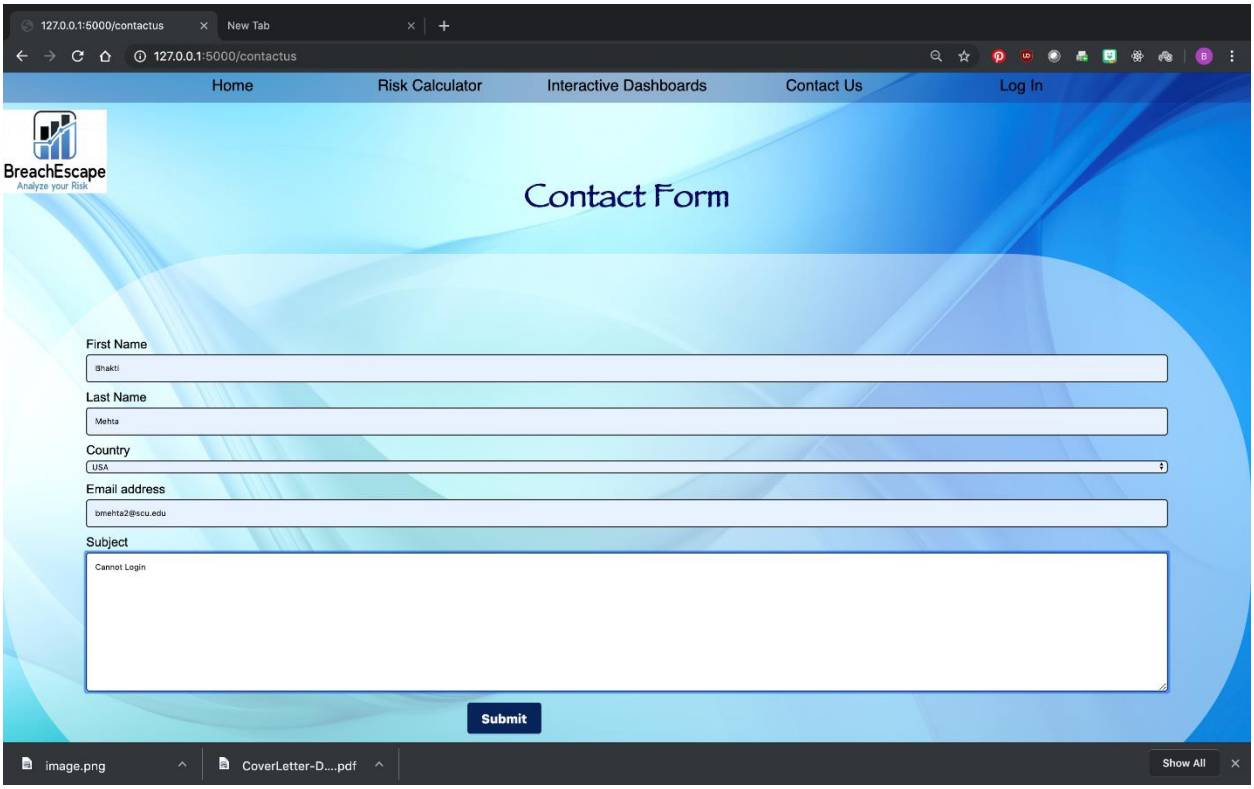
<table>
<tr><td colspan="2">Use Case Name: Logout of the BreachEscape account</td><td>Use case ID: 6</td><td>Importance level: High</td></tr>
<tr><td colspan="4">Primary Actor: 1. User/Customer/Company official/Security Solution provider
2. System/Admin</td></tr>
<tr><td colspan="4">Short Description: This describes how a registered user or system admin logs out of their BreachEscape Account</td></tr>
<tr><td colspan="4">Trigger: User wants to logout of their account
Type: External</td></tr>
<tr><td colspan="4">Preconditions: User is logged into their BreachEscape account</td></tr>
<tr><td colspan="2">Steps Performed:
1. User clicks on the Logout option from the menu bar.
2. User can see “You have successfully logged out”</td><td colspan="2">Information:
N/A</td></tr>
<tr><td colspan="4">Post Condition: User has successfully logged-out of their account and lands on the homepage.</td></tr>
<tr><td colspan="2">Major Inputs:
URL
Click on Menu items</td><td colspan="2">Major outputs
Description
Navigation through items
Page availability</td></tr>
</table>

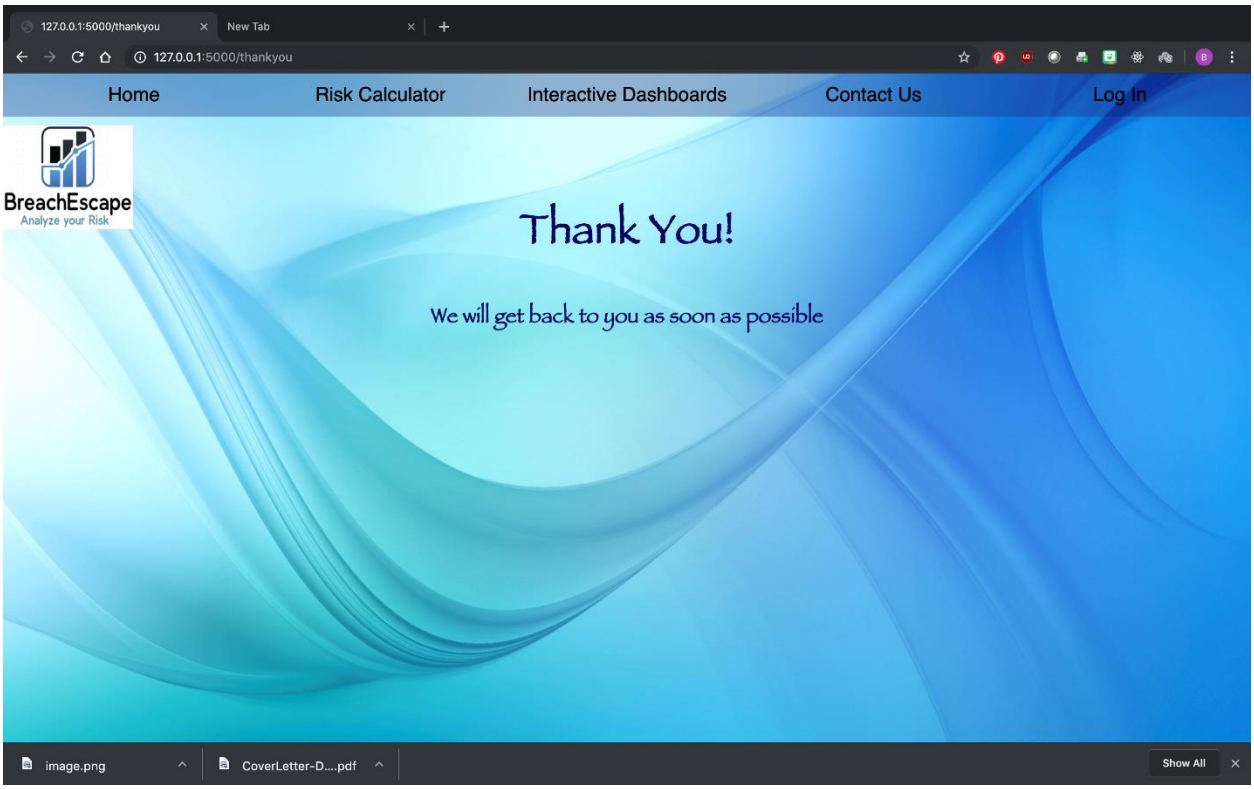


Use case 7:

Use Case Name: Contact us page	**Use case ID: 7**	*Importance level: High*
Primary Actor: 1. User/Customer/Company official/Security Solution provider 2. System/Admin		
Short Description: This describes how a guest or user can log a query with Breachescape.		
Trigger: User wants to log a query **Type:** External		
Preconditions: User knows the URL for Breachescape website		
Steps Performed: 1. User enters URL for Breachescape website. 2. Clicks on “contact us” on menu bar	**Information:** URL	

3. User Fills out the contact us form with their details and query. 4. User then clicks on submit button. 5. User is notified with “Thank you….we will get back to you as soon as possible”	User details Query Notification
Post Condition: User has successfully submitted his query and admin can see an entry in the database.	
Major Inputs: URL Click on Menu items User details	**Major outputs** Description Page availability Notification




127.0.0.1:5000/thankyou
New Tab
127.0.0.1:5000/thankyou
Home
Risk Calculator
Interactive Dashboards
Contact Us
Log In
BreachEscape
Analyze your Risk
Thank You!
We will get back to you as soon as possible
image.png
CoverLetter-D....pdf
Show All

12. Tools and Technology

Python
We used python as our programming language to write our scripts. All of the team members have taken a course on Data Science with Python and hence we used this technology for our project. We made use of environments like Canopy and Anaconda to work on Jupyter Notebooks for Python scripts. We also used the machine learning algorithms like Random Forest Classifier, Packages like sklearn, Feature selection methods and K-fold cross validation.

HTML, CSS
HTML was used to render the content on our web pages for the client-side application. We used CSS to add uniformity and well-defined structure across the web pages.

Flask: Here we have used python for training our machine learning model and also to create a web app. Flask is a python based microframework used for developing websites**.**
We have used Visual Studio Code as a platform for python scripting and for developing Web application using HTML and CSS.
We have created a home page, Risk assessment calculator, interactive dashboard page and contact page using HTML and CSS.

Tableau
We built tableau dashboards in order to host them on the Tableau Public Server. We built dashboards based on the available data to represent the descriptive analysis based on the past data and predictive analysis for the company using our risk assessment calculator. We also performed data blending where data was taken from 2 different sources and integrated before the visualization.

Database
We have used MySQL database to store and perform operations on the customer/user data. The Log in credentials would be saved in the database to perform user authentication.

13. Technical Approach

13.1 Data:

13.1.1 Data Preparation

We have used one survey dataset for our capstone project. This dataset is - Information security breaches survey, which was obtained from the official website of UK Government: data.gov.uk. This dataset is a general survey conducted by the UK government in 2012 consisting of information regarding number and type of security breaches affecting businesses. We have used dataset and its features to train our machine learning model for implementing the Risk Assessment Calculator.
Number of Observations: 3376 records
Source: Information Security Breaches survey from UK government website: data.gov.uk
Year of data collection: 2012

13.1.2 Data Dictionary:

Features	Description	Values
NumOT_virus_attacks_occured	**Number of times Virus Attacks occurred**	**Several Times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't Know**
NumOT_Data_Corruption_occured	**Number of times data corruption occured**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**

NumOT_staff_sent_inappropriate_emails	**Number of times staff sent inappropriate emails**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_staff_accessed_data_with_other_userID	**Number of times staff accessed data with some other user ID**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_computer_equipment_stolen	**Number of times computer equipment is stolen**	**Several Times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't Know**
NumOT_unauthorised_outsider_intercepted_communication	**Number of times unauthorised outsider intercepted communication**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_staff_broke_data_protection_laws	**Number of times staff broke data protection laws**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**

Unauthorised_outsider_tried_to_break_in	**Number of times unauthorised outsider tried to break in**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_identity_theft_occured	**Number of times identity theft occured**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_staff_misused_confidential_data	**Number of times staff misused the confidential data**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_phishing_occured	**Number of times phising occured**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_cofidential_data_stolen	**Number of times confidential data is stolen**	**Hundreds of times everyday** **Weekly** **Monthly** **A few times** **Only once** **Don't know**

NumOT_staff_accidently_lost/leak_confidential_data	**Number of times staff accidently lost confidential data**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**
NumOT_staff_used_computers_for_fraud	**Number of times staff used the computers for fraudulent activities**	**Weekly** **Monthly** **Few times** **Only once** **Don't know**
NumOT_staff_deliberately_sabotaged_data	**Number of times staff deliberately sabotaged data**	**Several times a day** **A few times** **Only once** **Don't know**
NumOT_staff_stole_computer_equipment	**Number of times staff stole computer equipment**	**Daily** **Weekly** **Monthly** **A few Times** **Only once** **Don't know**
Unauthorised_outsider_succeeded_in_penetrating	**Number of times unauthorised outsider succeeded in penetrating**	**Weekly** **Monthly** **Few times** **Only once** **Don't know**
NumOT_unauthorised_outsider_launced_DOS	**Number of times unauthorised outsider launched Denial of Service**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**

Business_sector	**Business Sector Levels**	**Banking and financial services** **Education** **Government** **Health** **IT** **Manufacturing and retail** **Real estate and Utilities** **Other**
ITexpenditure_on_security	**Amount spent on security**	**More than 50** **26-50** **11-25** **6-10** **2-5** **1 or less**
Priority_of_information_security	**Priority based on the information security**	**Very high** **High** **Neither High nor Low** **Low** **Not a priority** **Don't know**
Company_size	**Company Size classification**	**10,000+** **500-9999** **50-499** **Less than 50** **Don't Know**
Staff_understanding_of_security_policy	**Levels of staff understanding regarding security policy**	**Very well Understood** **Quite Well** **Poorly** **Don't know**

NumOT_Data_breach_occurred	**Number of times data breach occured**	**Hundreds of times a day** **Several times a day** **Daily** **Weekly** **Monthly** **A few times** **Only once** **Don't know**

13.1.3 Pre-processing the data

Our data cleaning process involved a lot of dataset modification and transformation in order to achieve data in desired format required for our analysis. We used NumPy and Pandas in Python to clean our dataset. The cleaning was performed using Jupyter Notebook and Canopy. Various cleaning measures included dropping unnecessary columns from the data frame, tidying up fields in the dataset, applying transformation functions to columns, handling the null data fields and so on.

13.1.3.1 Handling Redundant features

The dataset contains 25 attributes from 'Company size' to 'Number of times virus attacks occurred'. To fit the data into prediction model, we converted categorical values to numerical ones. Redundant columns like 'Type of role in the company' (similar to Business sector) were dropped.

13.1.3.2 Handling missing values

The dataset contained a few missing values which were filled with 0 using the following command

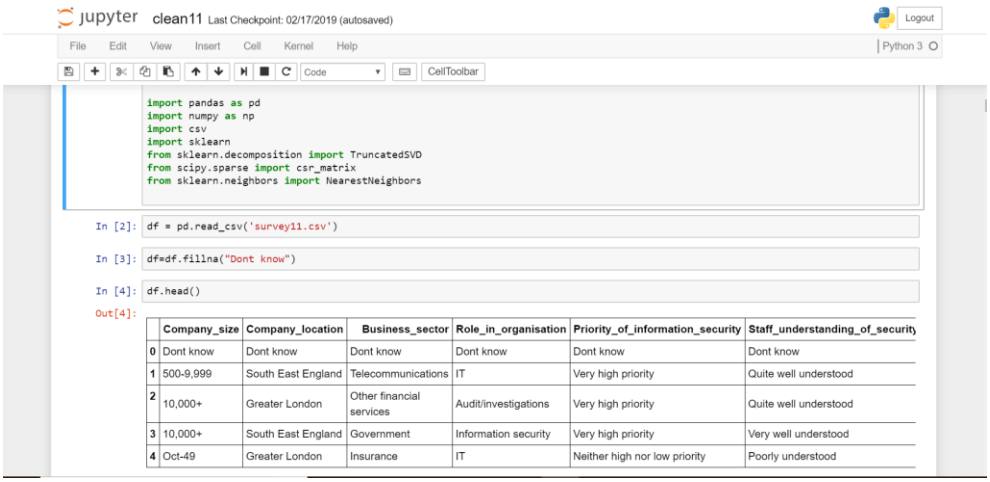
```
: df.fillna(0, inplace =True)
```

13.1.3 Scaling the data

The features in our dataset are categorical variables and do not have variant measurements and different magnitudes.

Therefore, Scaling is not needed in this case.

13.1.4 Transformation function



	Company_size	Company_location	Business_sector	Role_in_organisation	Priority_of_information_security	Staff_understanding_of_security
0	Dont know	Dont know	Dont know	Dont know	Dont know	Dont know
1	500-9,999	South East England	Telecommunications	IT	Very high priority	Quite well understood
2	10,000+	Greater London	Other financial services	Audit/investigations	Very high priority	Quite well understood
3	10,000+	South East England	Government	Information security	Very high priority	Very well understood
4	Oct-49	Greater London	Insurance	IT	Neither high nor low priority	Poorly understood

We used a general transformation function in python to create different categories for every column in the dataset. The algorithm for the function is as follows:

```
def function_name(x):
    If x == category1 :
        return 1
    elif x == category2 :
        return 2
    else :
        return 0
```

After applying the above mentioned function to every column, the field values of some features are shown below.

Column Name	Original value	Transformed value
Company size	10000 +	1
	500- 9999	2
	50-499	3
	Less than 10	4
	Don’t know	0

Column Name	Original value	Transformed value

Data breach occurred	Hundreds of times everyday / Several times a day	1
	Daily	2
	Weekly	3
	Monthly	4
	A few times	5
	Once only	6
	None	7
	Else	0

Column Name	Original value	Transformed value
Number of times	Several times a day	1
Virus attacks	Daily	2
occurred	Weekly	3
	Monthly	4
	A few times	5
	Once only	6
	None	7
	Else	0

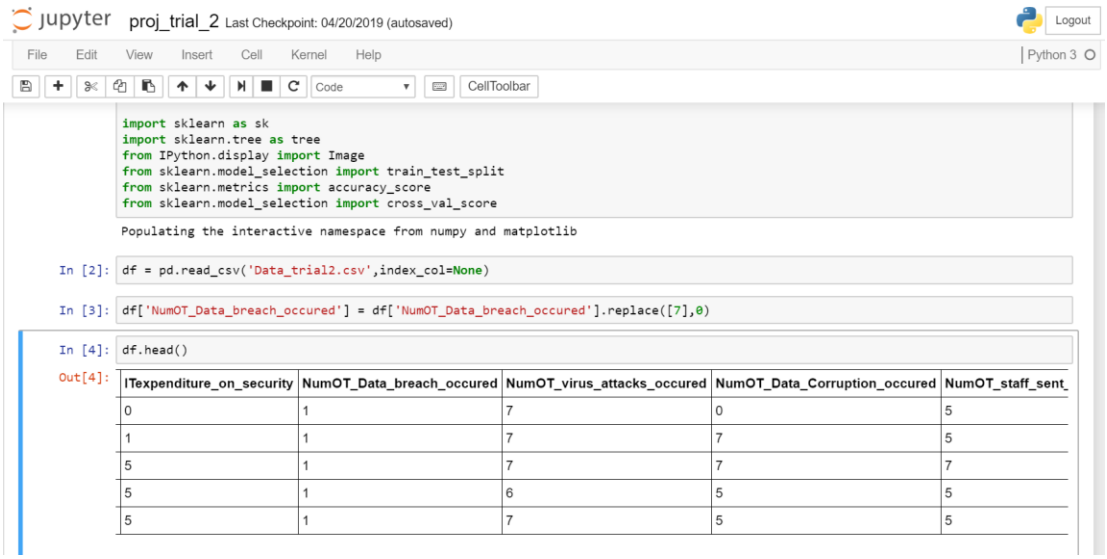
Column Name	Original value	Transformed value
Priority of	Very high priority	1
Information security	High priority	2

	Neither high nor low priority	3
	Low priority	4
	Not a priority at all	5
	Else	6

Similarly, all the feature values were transformed to numerical values between 0-6.

The probability prediction of our project is based on the feature 'Number of times data breach occurred'. In order to differentiate between events where breach occurred and did not occur, we grouped the values 1 – 6 as 1 and others as 0 for the attribute 'Number of times data breach occurred'

After preprocessing, the dataset looked as follows:



ITexpenditure_on_security	NumOT_Data_breach_occured	NumOT_virus_attacks_occured	NumOT_Data_Corruption_occured	NumOT_staff_sent_
0	1	7	0	5
1	1	7	7	5
5	1	7	7	7
5	1	6	5	5
5	1	7	5	5

13.1.5 Correlation with Target Feature:

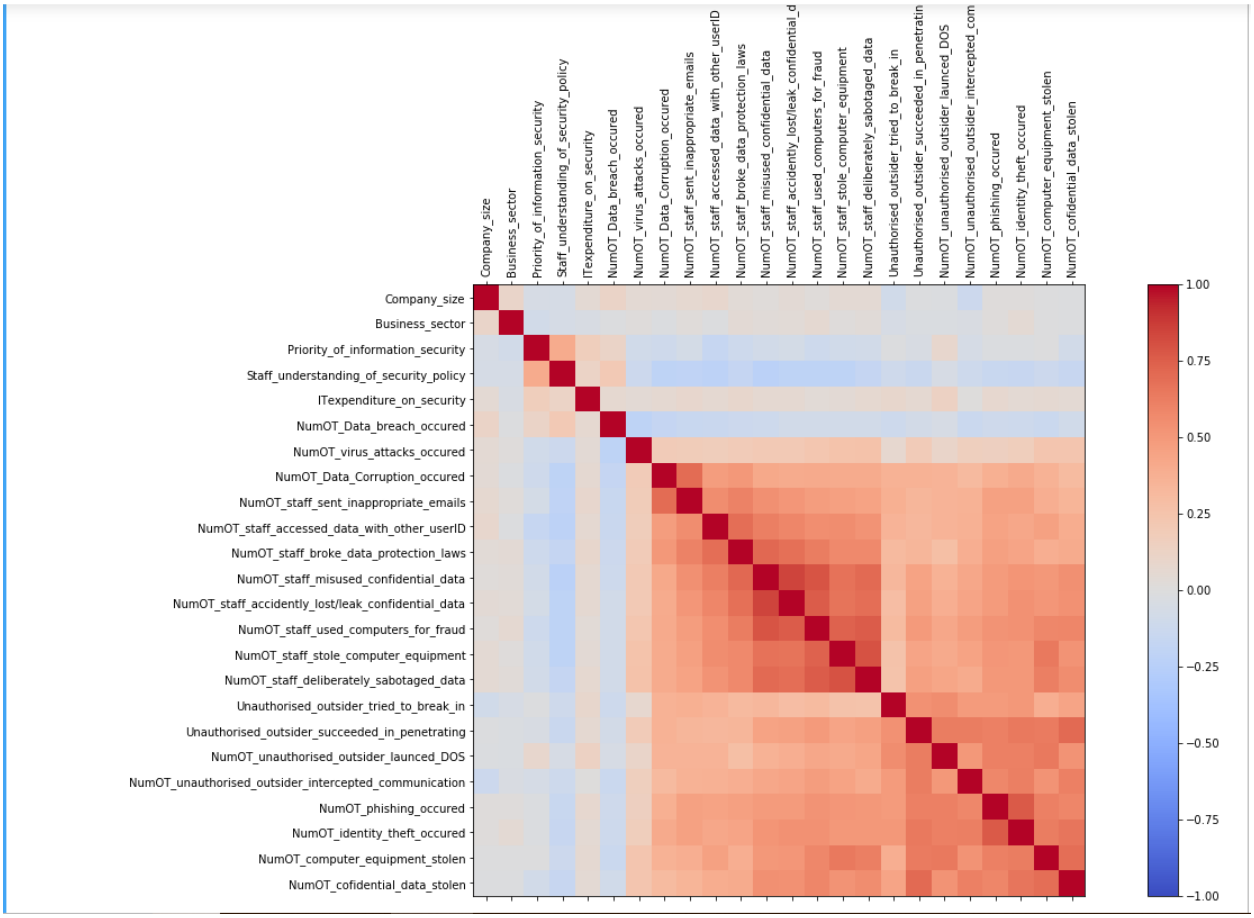
```
In [52]: corr = pd.DataFrame(df.corr()['NumOT_Data_breach_occured'])

In [54]: corr.to_csv('corr1_file.csv')
```

F15

	A	B
1	Features	NumOT_Data_breach_occured
2	NumOT_virus_attacks_occured	-0.203742369
3	NumOT_Data_Corruption_occured	-0.168365836
4	NumOT_staff_sent_inappropriate_emails	-0.141252664
5	NumOT_staff_accessed_data_with_other_userID	-0.140519366
6	NumOT_computer_equipment_stolen	-0.12832475
7	NumOT_unauthorised_outsider_intercepted_communication	-0.12641759
8	NumOT_staff_broke_data_protection_laws	-0.124110048
9	Unauthorised_outsider_tried_to_break_in	-0.112759757
10	NumOT_identity_theft_occured	-0.112384704
11	NumOT_staff_misused_confidential_data	-0.111138944
12	NumOT_phishing_occured	-0.103550391
13	NumOT_cofidential_data_stolen	-0.088334738
14	NumOT_staff_accidently_lost/leak_confidential_data	-0.084087389
15	NumOT_staff_used_computers_for_fraud	-0.083734808
16	NumOT_staff_deliberately_sabotaged_data	-0.083384451
17	NumOT_staff_stole_computer_equipment	-0.08000272
18	Unauthorised_outsider_succeeded_in_penetrating	-0.07747025
19	NumOT_unauthorised_outsider_launced_DOS	-0.050230255
20	Business_sector	-0.010424032
21	ITexpenditure_on_security	0.066437507
22	Priority_of_information_security	0.111498049
23	Company_size	0.113899875
24	Staff_understanding_of_security_policy	0.209003925
25	NumOT_Data_breach_occured	1

The increase in the value of Staff Understanding of Security Policy symbolizes decrease in the Staff Understanding . Thus, the Positive Correlation above actually represents an inverse relationship with the Number of Times Breach Occurred.
This Extends to all the other features as well.

Therefore,In our case, A positive Correlation represents Inverse Relationship and a Negative Correlation Represents a Direct Relationship with Number of Times Breach Occurred.



13.2 Building the model

In order to analyze the dataset, we built a prediction model on the survey data using different machine learning algorithms and classifiers, plot the results and calculated the accuracy of the model on the testing data. Based on the accuracy of the model and its fit on the data, we decided on which algorithm should be used for our analysis.

The three metrics to measure efficiency of the algorithms that we have used on our testing data in sklearn are :

1. **Accuracy:**

sklearn. metrics.accuracy_score(y_test, y_pred, normalize=True, sample_weight=None)

Here the calculation of accuracy_score by sklearn could be understood as follows - Given two lists or 1D arrays , y_test and y_pred , for every position index *i*, the *i-th* element of y_pred is compared with the *i-th* element of y_pred with the *i-th* element of y_test and perform the following calculation:

- Count the number of matches
- Divide it by the number of samples

Thus accuracy_score = number of matches/ number of samples

2. Receiver operating characteristic (ROC) curve

sklearn.metrics.roc_curve(y_test, y_score, pos_label=None, sample_weight=None, drop_intermediate=True)

On the ROC curve, typically, true positive rate is depicted on the Y axis, and false positive rate is on the X axis. This means that the top left corner of the plot is the "ideal" point - a false positive rate of zero, and a true positive rate of one. In order to compute FPR and TPR, the function *sklearn.metrics.roc_curve* is provided the true binary value and the target scores. After plotting the curve, we can say that larger area under the curve (AUC) is usually better.

3. Confusion matrix

*sklearn.metrics.confusion_matrix(*y_test, y_pred, labels=None, sample_weight=None*)*

A confusion matrix is a table that is used to describe the performance of a classification model on a set of test data for which the true values are known. It is used to tabulate the number of misclassifications, i.e., the number of predicted classes which were predicted wrong based on the true test classes.

In binary classification, the count of true negatives is C0,0, false negatives is C1,0, true positives is C1,1 and false positives is C0,1 where Ci,j is the total number of predictions

- True positives (TP): Cases where we predict Yes and the true outcome is Yes
- True negatives (TN): Cases where we predict no and the true outcome is No
- False positives (FP): Cases where we predict Yes but the true outcome is No (Also known as a "Type I error.")
- False negatives (FN): Cases where we predict No but the true outcome is Yes (Also known as a "Type I error.")

We had set of existing data that consisted of target values that we were aiming to predict. Thus we already had examples of right answers and just wanted to train our model to predict correctly based on it. Therefore we decided to select a number of Supervised Machine Learning algorithms to train our model and based on the accuracy metrics, decide which algorithm to work with.

Thus before moving on to fitting the model using the classifiers, we first sliced our data into two halves – Training data and Testing data. We divided the dataset in 90 – 10 % where test_size is 10% and train_size is 90% using the train_test_split() function of sklearn. X consisted of all the attributes in the dataset except the target variable and Y consisted of the target variable – 'Number of times data breach occurred'. Thus this splits the dataset into – X_train, y_train, X_test, y_test

```
In [6]: X = df.drop('NumOT_Data_breach_occured',axis=1)
        Y = df.NumOT_Data_breach_occured
        X_train, X_test, y_train, y_test = train_test_split( X, Y, test_size=0.1, random_state=42)
```

13.2.1 Decision Tree Classifier

class sklearn.tree.DecisionTreeClassifier(criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, r

andom_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split= None, class_weight=None, presort=False)

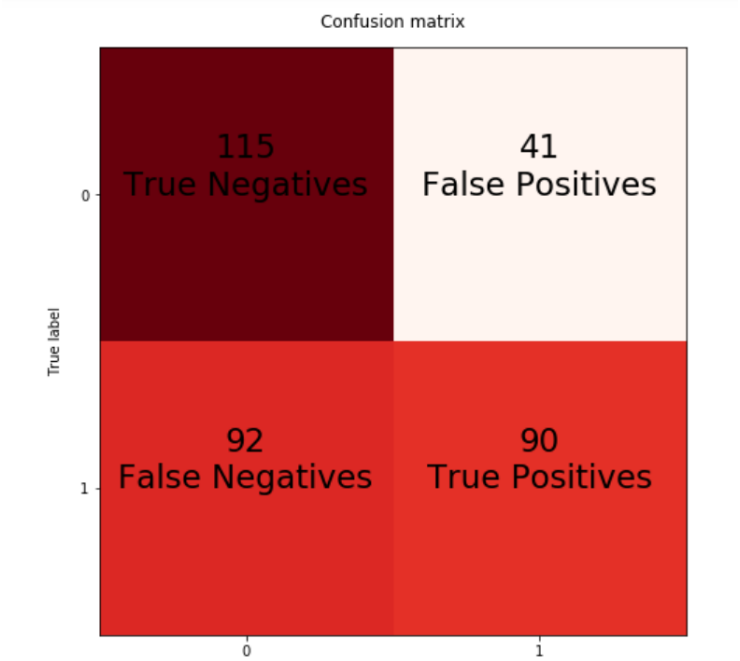
Decision Trees can be used as classifier or regression models. A dataset is broken down into smaller subsets based on a tree structure eventually leading us to the prediction. The prediction can be based on a binary model(1 or 0) or on a multiclass model. The algorithm follows a simple IF ElSE..AND..THEN logic down the nodes to bring us to the leaf node(the prediction). The root node (the first decision node) partitions the data based on the most influential feature in the dataset. There are 2 measures for this, Gini Impurity and Entropy. The goal of a decision tree classifier is to learn simple partitioning rules and predict the value of a target variable

```
In [7]: dt = tree.DecisionTreeClassifier(max_depth=2)
        dt.fit(X_train,y_train)
        y_pred_dt= dt.predict(X_test)
        print ('Accuracy of Decision Tree Classifier Model: ', accuracy_score(y_test,y_pred_dt))

        Accuracy of Decision Tree Classifier Model:  0.60650887574
```

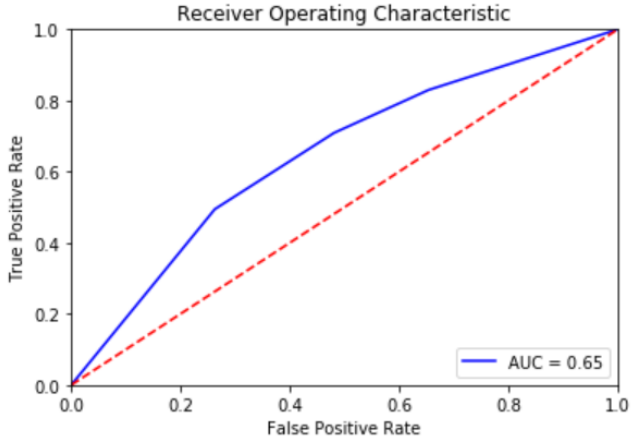
We fit our dataset to the classifier using DecisionTreeClassifier.fit() function. DecisionTreeClassifier.predict() function uses the X_test to predict the outcomes of the test data. Thus, the comparison between the true outcomes of y_test and y_pred_dt gives us the accuracy _score. The accuracy of our decision tree classifier was 60%. y_pred_dt gives us the binary classification of our target variable where 1 denotes the event where data breach occurred and 0 denotes the event where data breach did not occur.

```
In [13]: y_pred_dt

Out[13]: array([0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0,
                0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 1,
                0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0,
                0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0,
                0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0,
                0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0,
                1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0,
                0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0,
                0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1,
                0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0,
                1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1,
                0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 1,
                0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0,
                1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0], dtype=int64)
```

We further plotted the confusion matrix to get an idea of miscalculated predictions. The plot of the confusion matrix is as follows.



Thus we can see that the number of False positives(41) and False Negatives(92) is very high. We also implemented the ROC curve to understand the tradeoff between TPR and FPR. Classifiers that give curves closer to the top-left corner indicate a better performance. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test. Thus, from the ROC curve for our model we can see that the AUC is 0.65 and is quite close to the diagonal indicating high number of false positives. Thus reducing accuracy.



Parameter Tuning

- **Important Parameter :** To be tuned in a Decision Tree Model is the max_depth.
- **Reason for selecting this Hyper Parameter :** To avoid Overfitting of Data
- **Method Used : GridSearchCV**

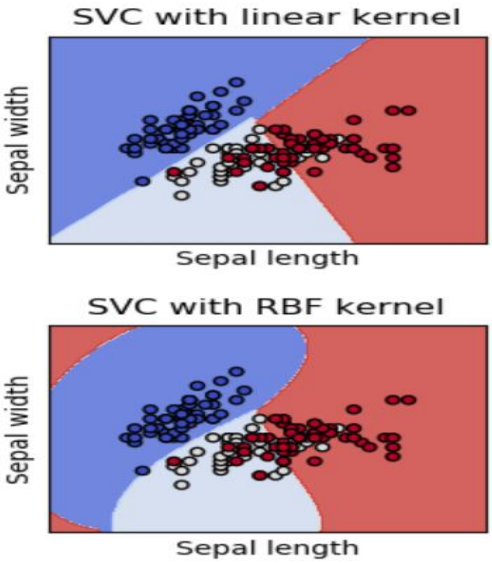
```
from sklearn.grid_search import GridSearchCV
def parameter_tuning(model,params):
   param_grid = params
   grid_model = model
   grid = GridSearchCV(grid_model, param_grid, cv=2, n_jobs=-1)
   grid.fit(X_train,y_train)
   return grid.best_estimator_
```

```
In [46]: dt_classifierModel = tree.DecisionTreeClassifier()
         param_grid = {'max_depth': [10, 20, 30, 40]}
         dt_best_estimator=parameter_tuning(dt_classifierModel,param_grid)
         dt_classifierModel = tree.DecisionTreeClassifier(max_depth=dt_best_estimator.max_depth)
         dt_classifierModel=dt_classifierModel.fit(X_train,y_train)
         dt_y_predictions = dt_classifierModel.predict(X_test)
         print ('Accuracy of Decision Tree Classifier after Parameter tuning : ', accuracy_score(y_test,dt_y_predictions))

         Accuracy of Random Forest Model after Parameter tuning :  0.985207100592

In [47]: dt_best_estimator.max_depth

Out[47]: 30
```

13.2.2 Support Vector Machine

class sklearn.svm.SVC(*C=1.0, kernel='rbf', degree=3, gamma='auto_deprecated', coef0=0.0, shrinking=True, probability=False, tol=0.001, cache_size=200, class_weight=None, verbose= False, max_iter=-1, decision_function_shape='ovr', random_state=None*)

Svm can be used for classification, regression as well as outlier detection problems. It is mostly used for classification problems. In a dataset, every data point is plotted in an n dimensional space where n is the number of features. The algorithm then defines a hyperplane (plane having 1 dimension less than the surround space) in such a way that the data points are classified into distinct classes. It may happen that the dataset can be sliced using various different hyperplanes. The hyperplane that creates maximum margin i.e maximum distance from data points of distinct classes should be chosen. The data points that are closer to the are called support vectors and they influence the position and orientation of the hyperplane.

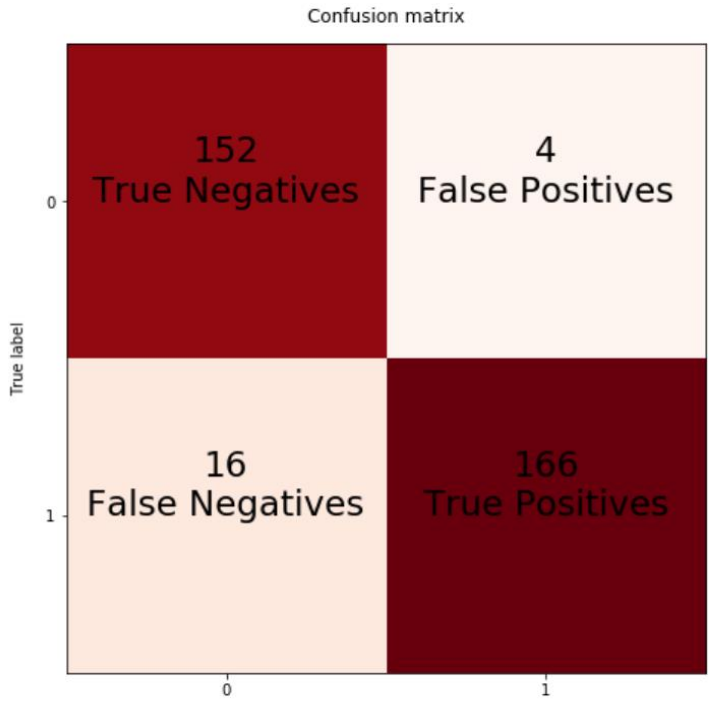


SVM classifier offers two types of kernels – linear and rbf. For linear kernel, the model builds the boundry between the classes based on some linear algebraic function of the type $a=b1+b2 \cdot X+b3 \cdot X2+b4 \cdot X3$. RBF uses normal curves around the data points, and sums these so that the decision boundary can be defined by a type of spacial condition such as curves.
The diagram below explains how the data points are classified using different types of hyperplanes(linear or curved) when the kernel is changed. SVM uses linear kernel by default.

```
In [11]: from sklearn.svm import SVC
         svclassifier = SVC(kernel='linear')
         svclassifier.fit(X_train, y_train)

Out[11]: SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
           decision_function_shape='ovr', degree=3, gamma='auto', kernel='linear',
           max_iter=-1, probability=False, random_state=None, shrinking=True,
           tol=0.001, verbose=False)

In [12]: y_pred_new = svclassifier.predict(X_test)
         print ('Accuracy of SVM: ', accuracy_score(y_test,y_pred_new))

         Accuracy of SVM:  0.612426035503
```

We fit our dataset to the classifier using svclassifier.fit()function. svclassifier.predict() function uses the X_test to predict the outcomes of the test data. Thus, the comparison between the true outcomes of y_test and y_pred_new gives us the accuracy _score. The accuracy of our SVM classifier was 61% with the linear kernel whereas it was 94% with rbf kernel. y_pred_new gives us the binary classification of our target variable where 1 denotes the event where data breach occurred and 0 denotes the event where data breach did not occur.

```
In [14]: from sklearn.svm import SVC
         svclassifier = SVC(kernel='rbf')
         svclassifier.fit(X_train, y_train)

Out[14]: SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
           decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
           max_iter=-1, probability=False, random_state=None, shrinking=True,
           tol=0.001, verbose=False)

In [15]: y_pred_new = svclassifier.predict(X_test)
         print ('Accuracy of SVM: ', accuracy_score(y_test,y_pred_new))

         Accuracy of SVM:  0.940828402367
```



We then plotted the confusion matrix for SVM with the Rbf kernel (as it returned results with higher accuracy) to get an idea of miscalculated predictions. The plot of the confusion matrix is as follows. The confusion matrix showed that the model returned 4 false positives and 16 false negatives

Parameter Tuning:

- **Important Parameter:** To be tuned in a Random Forest Model is Kernel and C and gamma.
- **C Parameter:** C is the the inverse of regularization strength in Logistic Regression.
- **Kernel Parameter:** Class of algorithms for pattern analysis.
- **Reason for selecting Kernel Hyper Parameter:** To recognize patterns in the data.
- **Reason for selecting C Parameter:** To improve the performance of the model on new, unseen data
- **Method Used: GridSearchCV**

```
from sklearn.grid_search import GridSearchCV
def parameter_tuning(model,params):
   param_grid = params
   grid_model = model
   grid = GridSearchCV(grid_model, param_grid, cv=2, n_jobs=-1)
   grid.fit(X_train,y_train)
   return grid.best_estimator_
```

```
SVM_classifierModel = SVC()
param_grid = {'kernel': ['linear','rbf','poly'],'C': [1,10,100,1000]}
SVM_best_estimator=parameter_tuning(SVM_classifierModel,param_grid)
SVM_classifierModel = SVC( kernel=SVM_best_estimator.kernel, C=SVM_best_estimator.C)
SVM_classifierModel=SVM_classifierModel.fit(X_train,y_train)
SVM_classifier_predictions = SVM_classifierModel.predict(X_test)
print ('Accuracy of Random Forest Model after Parameter tuning : ', accuracy_score(y_test,SVM_classifier_predictions))
```

```
('Accuracy of Random Forest Model after Parameter tuning : ', 0.9970414201183432)
```

```
SVM_best_estimator.kernel
```

```
'rbf'
```

```
SVM_best_estimator.C
```

```
100
```

13.2.3 Logistic Regression

class sklearn.linear_model.LogisticRegression(penalty='l2', dual=False, tol=0.0001, C=1.0, fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='warn', max_iter=100, multi_class='warn', verbose=0, warm_start=False, n_jobs=None, l1_ratio=None)

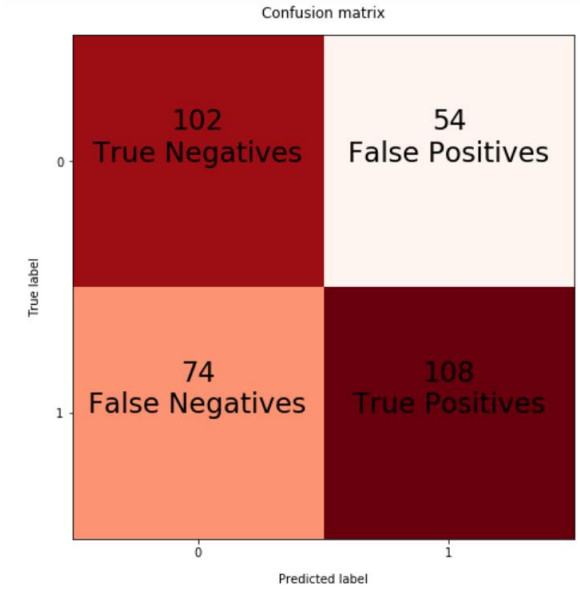
Logistic Regression is used for classification problems where the predicted outcome is binary(yes/no), ordinal or categorical. The logistic function, also called the sigmoid function is an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

$$G(x) = 1 / (1 + e^{-value})$$

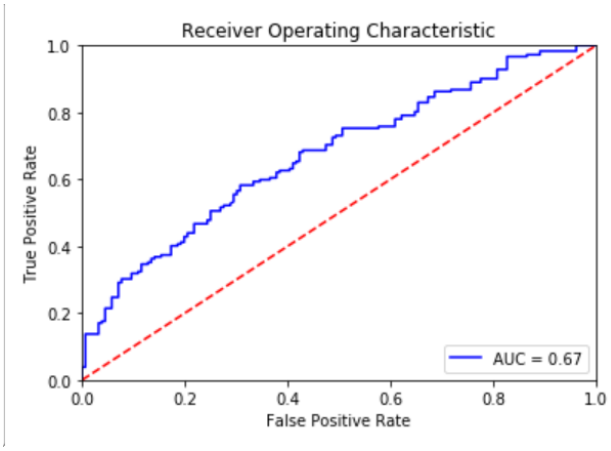
Thus the sigmoid curve spans between the datapoints in such a way that we can classify them into distinct binary classes based on a decision boundary.

```
In [150]: from sklearn.linear_model import LogisticRegression
          logisticRegr = LogisticRegression()
          logisticRegr.fit(X_train, y_train)
          y_pred_lr= logisticRegr.predict(X_test)
          print ('Accuracy of Logistic regression: ', accuracy_score(y_test,y_pred_lr))

          Accuracy of Logistic regression:  0.621301775148

In [151]: y_pred_lr

Out[151]: array([0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1,
                 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1,
                 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0,
                 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1,
                 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1,
                 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0,
                 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1,
                 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0,
                 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
                 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0,
                 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1,
                 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1,
                 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0,
                 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1,
                 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0], dtype=int64)
```

We fit our dataset to the classifier using LogisticRegression.fit()function. LogisticRegression.predict() function uses the X_test to predict the outcomes of the test data. Thus, the comparison between the true outcomes of y_test and y_pred_lr gives us the accuracy _score. The accuracy of our Logistic regression classifier was 62%. y_pred_lr gives us the binary classification of our target variable, where 1 denotes the event where data breach occurred and 0 denotes the event where data breach did not occur.



After plotting the confusion matrix we can see that our model has predicted 54 positives and 74 negatives falsely. We also implemented the ROC curve to understand the relationship between TPR and FPR. From all these efficiency metrics we can say that Logistic Regression does not seem to be the best choice for a prediction model for our dataset.



Parameter Tuning:

- **Important Parameter:** To be tuned in a Random Forest Model is C and Penalty parameter.
- **C Parameter:** C is the the inverse of regularization strength in Logistic Regression.
- **Penalty Parameter:** Defines the penalty to be added to the loss function. It can be Lasso or Ridge.

- **Reason for Penalty Hyper Parameter:** To avoid Overfitting and Underfitting of Data
- **Reason for selecting C Parameter:** To improve the performance pf the model on new, unseen data
- **Method Used: GridSearchCV**

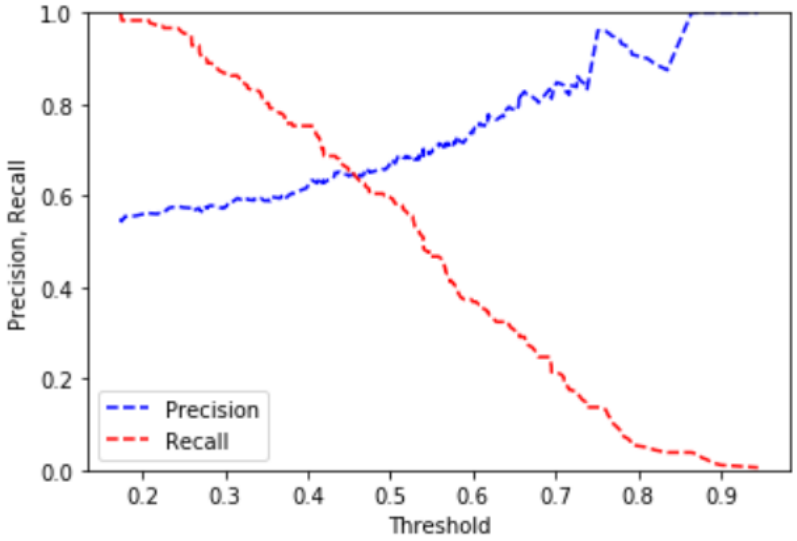
```
from sklearn.grid_search import GridSearchCV
def parameter_tuning(model,params):
   param_grid = params
   grid_model = model
   grid = GridSearchCV(grid_model, param_grid, cv=2, n_jobs=-1)
   grid.fit(X_train,y_train)
   return grid.best_estimator_
```

```
79]: from sklearn.linear_model import LogisticRegression
     lr_classifierModel = LogisticRegression()
     param_grid = {'C': [1,10,100,1000],'penalty' : ['l1', 'l2']}
     lr_best_estimator=parameter_tuning(lr_classifierModel,param_grid)
     lr_classifierModel = LogisticRegression(C=lr_best_estimator.C,penalty=lr_best_estimator.penalty)
     lr_classifierModel=lr_classifierModel.fit(X_train,y_train)
     lr_y_predictions = lr_classifierModel.predict(X_test)
     print ('Accuracy of Logistic Regression Model after Parameter tuning : ', accuracy_score(y_test,lr_y_predictions))

     Accuracy of Logistic Regression Model after Parameter tuning :  0.599112426036

80]: lr_best_estimator.C

80]: 1

81]: lr_best_estimator.penalty

81]: 'l2'
```

Selecting the Probability Threshold:

The Probability Threshold Selected is 0.45
The Probability is selected by taking a Tradeoff between Recall and Threshold as shown Below.

```
Out[67]: (0, 1)
```



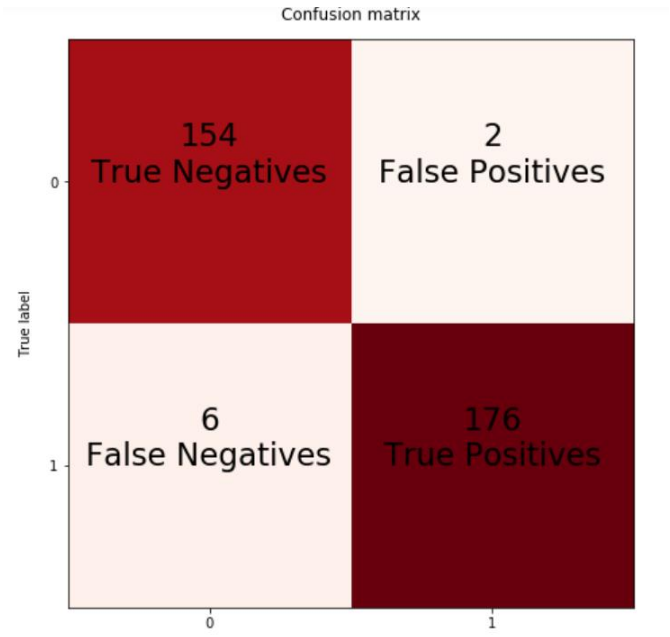
13.2.4 Random Forest

class sklearn.ensemble.RandomForestClassifier(n_estimators='warn', criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None)

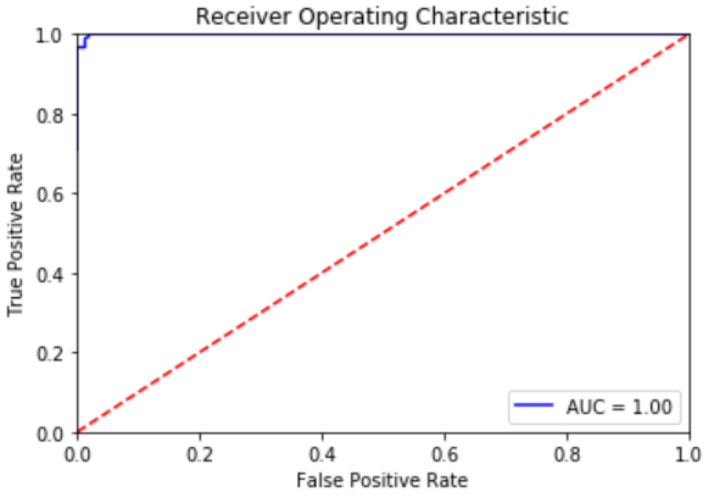
Random forest classifier divides the training dataset to form subsets in order to form multiple decision trees that individually return predicted classes of the decision variable. In an RF classifier, higher the number of trees in the forest gives higher accuracy results. The RF classifier selects *i* number of features from total of *j* features such that $i < j$. Among the *i* features, calculate node x using the best point where the data could be split. This is the root node. Keep splitting the dataset to form smaller trees thus gaining y nodes. As we repeat all of the above steps for n number of times, we gain n number of trees. Outcomes of all the decision trees are then aggregated to predict one class for every data point. This gives a more accurate result than decision tree classifier.

```
In [154]: from sklearn.ensemble import RandomForestClassifier
          rfmodel = RandomForestClassifier()
          rfmodel.fit(X_train,y_train)
          y_pred_rf = rfmodel.predict(X_test)
          print ('Accuracy of Random Forest Classifier Model: ', accuracy_score(y_test,y_pred_rf))

          Accuracy of Random Forest Classifier Model:  0.976331360947

In [155]: y_pred_rf

Out[155]: array([0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1,
                 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0,
                 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1,
                 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1,
                 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0,
                 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0,
                 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0,
                 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0,
                 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1,
                 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0,
                 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1,
                 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0,
                 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0,
                 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0,
                 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0], dtype=int64)
```

We fit our dataset to the classifier using RandomForestClassifier.fit()function. RandomForestClassifier.predict() function uses the X_test to predict the outcomes of the test data. Thus, the comparison between the true outcomes of y_test and y_pred_rf gives us the accuracy _score. The accuracy of our Random Forest classifier was 97%. This was the highest accuracy that any classification model had returned so far. y_pred_rf gives us the binary classification of our target variable, where 1 denotes the event where data breach occurred and 0 denotes the event where data breach did not occur.



The confusion matrix that we plotted for RF showed that the classifier predicted only 2 positives and 6 negatives falsely. This classifier was so far the best fit for our dataset



The ROC plot also returned a near perfect curve with maximum area under curve ratio of 1. The curve was most closest to the upper left hand corner which signifies maximum True Positive rate and minimum false positive rate.

Parameter Tuning:

- **Important Parameter:** To be tuned in a Random Forest Model is the max_depth.
- **Reason for selecting this Hyper Parameter:** To avoid Overfitting of Data
- **Method Used: GridSearchCV**

```
from sklearn.grid_search import GridSearchCV
def parameter_tuning(model,params):
   param_grid = params
   grid_model = model
   grid = GridSearchCV(grid_model, param_grid, cv=2, n_jobs=-1)
   grid.fit(X_train,y_train)
   return grid.best_estimator_
```

```
]: rf_classifierModel = RandomForestClassifier(n_estimators=100, n_jobs=-1)
   param_grid = {'max_depth': [10, 20, 30, 40]}
   rf_best_estimator=parameter_tuning(rf_classifierModel,param_grid)
   rf_classifierModel = RandomForestClassifier(n_estimators=100, n_jobs=-1, max_depth=rf_best_estimator.max_depth)
   rf_classifierModel=rf_classifierModel.fit(X_train,y_train)
   rf_y_predictions = rf_classifierModel.predict(X_test)
   print ('Accuracy of Random Forest Model after Parameter tuning : ', accuracy_score(y_test,rf_y_predictions))

   Accuracy of Random Forest Model after Parameter tuning :  0.994082840237

]: max_depth=rf_best_estimator.max_depth

]: max_depth

]: 40
```

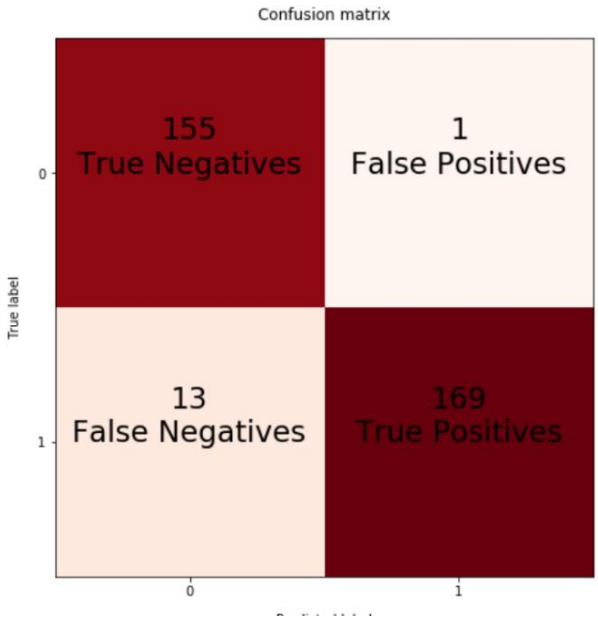
13.2.5 Ensemble model

class sklearn.ensemble.VotingClassifier(estimators, voting='hard', weights=None, n_jobs=None , flatten_transform=True)

Ensemble model aims at combining the predictions of several base estimators built with a given learning algorithm in order to improve accuracy of model over a single estimator. Voting is one of the three methods in which ensemble model can be built (Bagging and Boosting being the other two). In averaging/ voting methods, the main idea is to build several estimators independently and then to average their predictions.

```
In [158]: from sklearn.ensemble import VotingClassifier
          ensemble_model = VotingClassifier(estimators=[('rf', model), ('lr', logisticRegr), ('svm' , svclassifier)], voting='hard')
          ensemble_model.fit(X_train,y_train)
          ensemble_model.score(X_test,y_test)

Out[158]: 0.95857988165680474

In [159]: y_pred_em = ensemble_model.predict(X_test)

In [160]: y_pred_em

Out[160]: array([0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1,
                 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0,
                 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1,
                 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1,
                 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0,
                 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0,
                 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0,
                 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0,
                 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1,
                 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0,
                 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1,
                 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0,
                 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0,
                 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0,
                 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0], dtype=int64)
```

Thus here we have used Random Forest classifier, Logistic Regression Classifier and Support vector machine in combination to classify target variable using voting method. We fit our dataset to the classifier using ensemble_model.fit() function. Ensemble_model.predict() function uses the X_test to predict the outcomes of the test data. The accuracy of our Ensemble model was 95%



The confusion matrix that we plotted for Ensemble model showed that the classifier predicted 12 positives and 13 negatives falsely.

Thus taking into consideration all of the above algorithms, their accuracies and fit on our dataset, we have decided to train our model using Random Forest Classifier

Comparison of all Models:

Model	Accuracy	Precision	Recall	F-1 Score
Decision Tree	0.6	0.68	0.49	0.57
SVM with Linear Kernel	0.61	0.67	0.53	0.59
SVM with Rbf Kernel	0.94	0.97	0.91	0.94
Logistic Regression	0.62	0.66	0.59	0.62
Random Forest	0.98	1	0.97	0.98

13.3 Model evaluation

K Fold Cross Validation

class sklearn.model_selection.KFold(*n_splits='warn', shuffle=False, random_state=None*)

Cross-validation is a resampling process used to evaluate machine learning models on a dataset. The dataset is split into k partitions. For every iteration, one partition is maintained as test data set and remaining partitions(k-1) are used as training data. The model is then fitted on the training partition and a score is returned. Scores for all the partitions are aggregated to give performance validation of the model. Here we have split our dataset into 10 partitions.

```
In [172]: print(cross_val_score(rfmodel, X, Y, cv=10, scoring='accuracy').mean())

          0.99230064807
```

Thus the Random Forest model returned an accuracy of 99% after K -fold cross validation. This is a very high accuracy and thus confirms that this model is suitable for our analysis.

13.4 Parameter Tuning

Parameter tuning is also called as Hyperparameter optimization where the algorithm parameters are referred to as hyperparameters.
Parameter tuning for Random Forest involves working with mainly two parameters – the number of decision trees formulated by the algorithm and the depth of the decision trees. The number of decision trees is denoted by *n_estimators* whereas the depth is denoted by *max_depth.* Thus max_depth represents the depth of each tree in the forest. The deeper the tree, the more splits it has and it captures more information about the data.

```
In [173]: rf_classifierModel = RandomForestClassifier(n_estimators=100, n_jobs=-1)
          param_grid = {'max_depth': [10, 20, 30, 40]}
          rf_best_estimator=parameter_tuning(rf_classifierModel,param_grid)
          rf_classifierModel = RandomForestClassifier(n_estimators=100, n_jobs=-1, max_depth=rf_best_estimator.max_depth)
          rf_classifierModel=rf_classifierModel.fit(X_train,y_train)
          rf_y_predictions = rf_classifierModel.predict(X_test)
          print ('Accuracy of Random Forest Model after Parameter tuning : ', accuracy_score(y_test,rf_y_predictions))

          Accuracy of Random Forest Model after Parameter tuning :  0.979289940828

In [174]: max_depth=rf_best_estimator.max_depth

In [175]: max_depth

Out[175]: 40
```

Thus after fitting the model on different the parameter of max depth of 10, 20,30 and 40, we found that RF classifier for our dataset works best with max depth of 40.

13.5 Feature Selection

class sklearn.feature_selection.RFE(estimator, n_features_to_select=None, step=1, verbose=0)

Datasets contain very large number of features. Some of those features might be unnecessary for the analysis. An important point is to decide which features to use. There are an infinite number of combinations possible. Thus we need to pick out a combination of features that gives highest prediction accuracy.

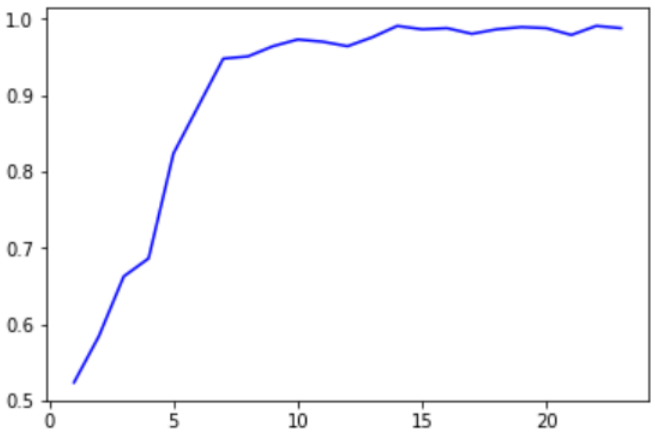
The aim of recursive feature elimination (RFE) is to select features using an external estimator that assigns weights to features by recursively considering smaller and smaller sets of features.
First, the estimator is trained on the initial set of features and the importance of each feature is obtained through a *feature_importances_* attribute. Then, the least important features are pruned from current set of features. This procedure is repeated until we reach a number of features that gives maximum accuracy.
Feature importances for the RF model are as follows

```
In [176]: rfmodel.feature_importances_

Out[176]: array([ 0.06,  0.09,  0.06,  0.05,  0.07,  0.09,  0.05,  0.06,  0.05,
                  0.03,  0.02,  0.03,  0.02,  0.03,  0.01,  0.06,  0.02,  0.03,
                  0.03,  0.04,  0.04,  0.03,  0.01])
```

0	Company_size	1
1	Business_sector	2
2	Priority_of_information_security	3
3	Staff_understanding_of_security_policy	4
4	ITexpenditure_on_security	5
5	NumOT_virus_attacks_occured	6
6	NumOT_Data_Corruption_occured	7
8	NumOT_staff_accessed_data_with_other_userID	8
9	NumOT_staff_broke_data_protection_laws	9
7	NumOT_staff_sent_inappropriate_emails	10
10	NumOT_staff_misused_confidential_data	11
12	NumOT_staff_used_computers_for_fraud	12
11	NumOT_staff_accidently_lost/leak_confidential_...	13
15	Unauthorised_outsider_tried_to_break_in	14
14	NumOT_staff_deliberately_sabotaged_data	15
13	NumOT_staff_stole_computer_equipment	16
17	NumOT_unauthorised_outsider_launced_DOS	17
18	NumOT_unauthorised_outsider_intercepted_commun...	18
16	Unauthorised_outsider_succeeded_in_penetrating	19

After using the RFE estimator, our model returns maximum accuracy of 99% at 19 features. We print this list of 19 features called *sorted_features* along with their ranks.

Thus, if we plot a graph of length of features against accuracy, after applying the feature selection RFE estimator, we can see that after rising up till a certain point, graph flattens out. This exact point is the maximum number of features that needs to be included to give maximum accuracy. In our case it is 19.



13.6 Building a new model:

After completion of the feature selection process, we had 19 features on which we needed to build and fit our Random Forest Classifier model. Hence we created a new dataframe called new_df containing the top 19 features based on their feature importances.

We again had to define new X and Y split on this new dataframe to divide the data into training and testing sections. We divided the dataset in 90 – 10 % where test_size is 10% and train_size is 90% using the train_test_split() function of sklearn. X consisted of all the attributes in the dataset except the target variable and Y consisted of the target variable – 'Number of times data breach occurred'. Thus this splits the dataset into – X_new_train, y_new_train, X_new_test, y_new_test

```
In [187]: X_new = new_df.drop('NumOT_Data_breach_occured',axis=1)
          Y_new = new_df.NumOT_Data_breach_occured
          X_new_train, X_new_test, y_new_train, y_new_test = train_test_split( X_new, Y_new, test_size=0.1, random_state=42)

In [188]: from sklearn.ensemble import RandomForestClassifier
          new_rfmodel = RandomForestClassifier()
          new_rfmodel.fit(X_new_train,y_new_train)
          new_y_pred_rf = new_rfmodel.predict(X_new_test)
          print ('Accuracy of Random Forest Classifier Model: ', accuracy_score(y_new_test,new_y_pred_rf))

          Accuracy of Random Forest Classifier Model:  0.991124260355
```

We then implemented the Random Forest Classifier on this new dataset. We fit our dataset to the classifier using RandomForestClassifier.fit()function. RandomForestClassifier.predict() function uses the X_new_test to predict the outcomes of the test data. Thus, the comparison between the true outcomes of y_new_test and new_y_pred_rf gives us the accuracy _score. Now the accuracy of our Random Forest classifier is 99%. Thus we can see a rise of 2% in the prediction accuracy of the model after feature engineering.

13.7 Fitting the model

Thus after deciding on the classifier to use and the features to use it on, we saved this model to use with unknown data (user entered data) for which we need to predict outcomes. For this we made use of pickle in python which is a powerful algorithm for serializing and de-serializing a Python object structure. The algorithm returns an accuracy score using new X_test and new y_test. The accuracy of this pickled model is 99%.

```
In [190]: import pickle

In [191]: filename='Final_model.pkl'

In [192]: pickle.dump(new_rfmodel,open(filename,'wb'),protocol=2)

In [193]: loaded_model=pickle.load(open(filename,'rb'))

In [195]: result=loaded_model.score(X_new_test,y_new_test)

In [196]: result
Out[196]: 0.99112426035502954
```

14. Creating Web Application using Flask

Here we have used python for training our machine learning model and also to create a web app. Flask is a python based microframework used for developing websites.
HTML/CSS pages: We have used Visual Studio Code as a platform for python scripting and for developing Web application using HTML and CSS.
We have created a home page, Risk assessment calculator, interactive dashboard page and contact us page using HTML and CSS.
We have created a form page for Risk assessment calculator with 19 fields that are mapped as follows: (Based on the Data Dictionary mentioned in Data Preparation)

```
<div class="container">
    <form action="/result" id="survey-form" method="POST">
        <div class="labels">
            <label id="" for="Comp_sz">Company size</label>
        </div>
        <div class="input-tab">
            <select id="dropdown" name="Comp_sz">
                <option value="0">Unknown</option>
                <option value="1">10,000+</option>
                <option value="2">500 - 9,999</option>
                <option value="3">50 - 499</option>
                <option value="4">Less than 10</option>
            </select>
        </div>
        <div class="labels">
            <label id="Business_sec" for="business_sector">Business Sector</label>
        </div>
        <div class="input-tab">
            <select id="dropdown" name="Business_sec">
                <option value="0">Other</option>
                <option value="1">Banking and Financial services</option>
                <option value="2">Education</option>
                <option value="3">Government</option>
                <option value="4">Health</option>
                <option value="5">IT</option>
                <option value="6">Manufacturing and Retail</option>
                <option value="7">Real estate and Utilities</option>
```

14.1 Flask Documentation and framework

A virtual environment needs to be created in the PowerShell of Visual Studio code to manage the dependencies in the project, both in development and in production.
http://flask.pocoo.org/docs/1.0/installation/#installation

1.Creating virtual environment

Create a project folder and a venv folder within:

mkdir myproject
cd myproject
Python -m venv venv

2.If you are using python 2 instead of python 3

sudo python2 Downloads/get-pip.py
sudo python2 -m pip install virtualenv

Python 3 comes bundled with the venv module to create virtual environments. If you're using a modern version of Python, you can continue on to the next section.

3.Activating Environment

Before beginning the project, activate the corresponding environment
.venv/bin/activate
On Windows:
venv\Scripts\activate
Your shell prompt will change to show the name of the activated environment.

4. To activate python consider:

python

Instead of Power Shell it will Change to Python

5.Install and Run Flask

Within the activated environment, use the following command to install Flask:

pip install Flask

Once activated and Flask is installed to run flask use:

python -m flask run

6.Create script.py file in the project folder and view the content in the file below:

Here we import the libraries and packages like: os, numpy, sklearn.ensemble.forest, flask, pickle, then using *app=Flask('__name__')* we create an instance of flask. *@app.route('/')* is used to tell flask what URL should trigger the functions home, Valuepredictor, signup, loggedin, navLogin, result, test, contactus, thankyou, successful registration, confirmuser and forgot_password.

script.py

```
import os
import numpy as np
import sklearn.ensemble.forest
import flask
import pickle
from flask import Flask, render_template, request
from flask_mail import Mail
from dbconnect import connection
from flask_mail import Message
```

In the functions we use *render_template('function_name.html')* to display the webpage in the browser**.**

```
#to tell flask what url shoud trigger the function inde
# x()
@app.route('/')
@app.route('/home',methods = ['POST','GET'])
def home():
    return flask.render_template('home.html',is_logged_in=is_logged_in)
```

7.Running the application:

export FLASK_APP=script.py

Python -m flask run

When someone submits the form, the webpage should display the predicted value as Risk Score. For this, we require the model file *(Final_model.pkl)* we created before, in the same project

folder. This integration of the pickled file-Python script and UI can be seen in Script.py file in the repository.

Here after the form is submitted, the form values are stored in variable *to_predict_list* in the form of dictionary. We convert it into a list of the dictionary's values and pass it as an argument to *ValuePredictor()* function**.**

```
@app.route('/result',methods = ['POST'])
def result():
    if request.method == 'POST':
        to_predict_list = request.form.to_dict()
        to_predict_list=list(to_predict_list.values())
        to_predict_list = list(map(int, to_predict_list))
        result = ValuePredictor(to_predict_list)
        result =result[1]
        risk_score = round(result,2)
        if (round(result,2)>=0 and (round(result,2)<=0.3)):
             prediction=' Your Risk Level : Minimal'
        elif (round(result,2)>0.3) and (round(result,2)<=0.5):
            prediction='Your Risk Level : Moderate'
        elif (round(result,2)>0.5) and (round(result,2)<=0.65) :
            prediction='Your Risk Level : High'
        elif (round(result,2)>0.65) and (round(result,2)<=0.85) :
            prediction='Your Risk Level : Critical'
        elif (round(result,2)>0.85) and (round(result,2)<=1) :
            prediction='Your Risk Level : Catastrophic'

    return render_template("result.html",prediction=prediction,risk_score=risk_score)
```

In *ValuePredictor()* function, we load the *Final_model.pkl* file and predict the new values and return the result.

```
#prediction function
def ValuePredictor(to_predict_list):
    to_predict = np.array(to_predict_list).reshape(1,19)
    loaded_model = pickle.load(open("Final_model.pkl","rb"))
    result = loaded_model.predict_proba(to_predict)
    return result[0]
```

This result/prediction(Risk score) is then passed as an argument to the template engine with the result- html page to be displayed.

8. Run the application again using

Python -m flask run

and it should predict the Risk Score after submitting the form.

15. GAP Analysis And Solutions

Current Scenario:
Currently few companies like Forbes, Cisco, Aon, Bitsight, Mindtools provide risk analysis reports and solutions to the companies who have sensitive information and are susceptible to the breach. These solutions which are available in the market are either very expensive or not very accurate. These facilities also sometimes need the users to sign up for an account to access the data analytics. We at BreachEscape provide a free and open platform for all the users to gain benefits of our risk calculators. We provide accurate results due to our up-to-date data sets used for analysis of risks and data breaches.

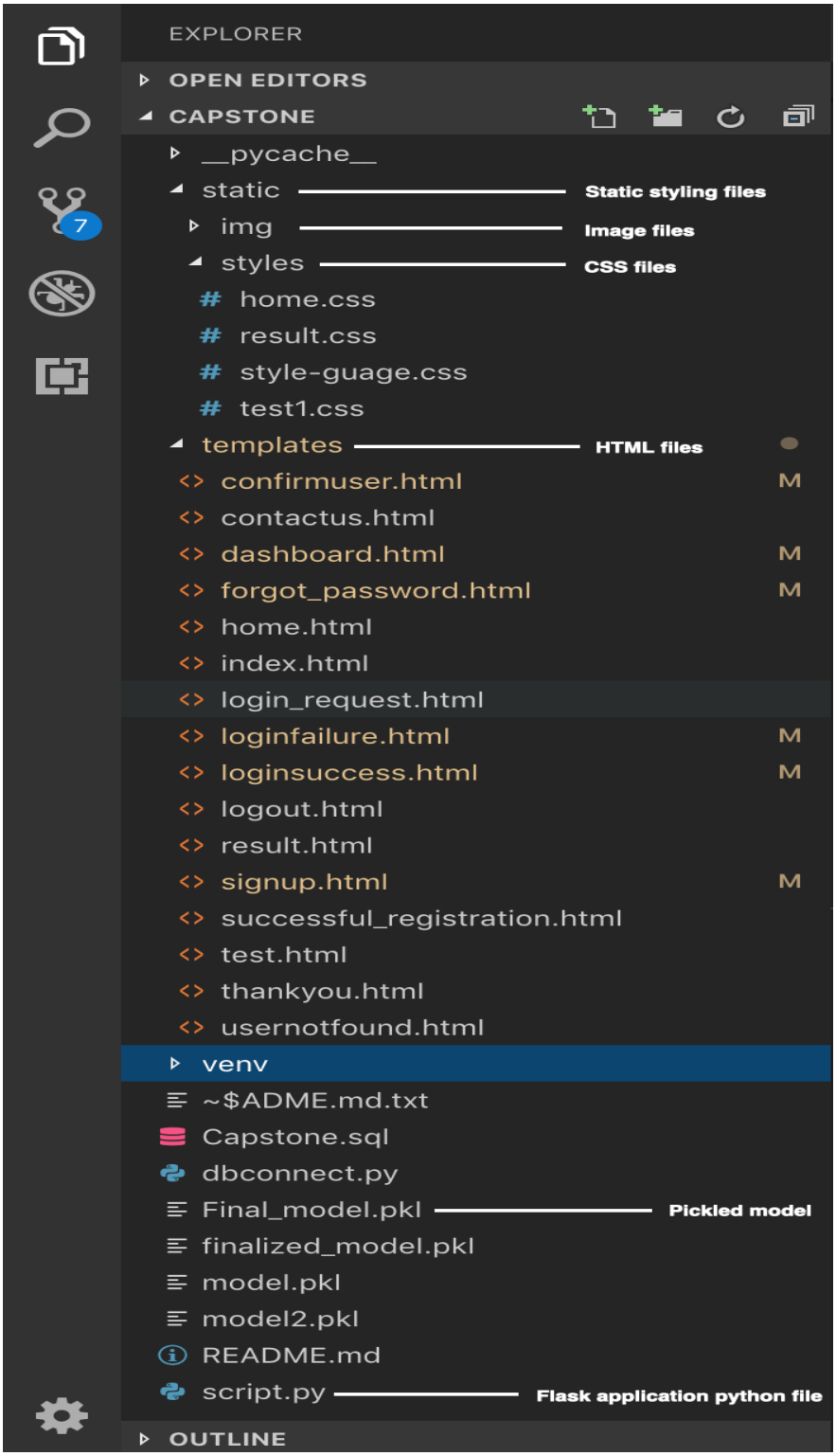
One-stop solution:
There are multiple dimensions to data-risk analytics. There are thousands of vulnerabilities that can affect the company's data. The solutions offered today, are not comprehensive enough and do not consider the various aspects of data-breach analysis.
Thus, our solution offers a one-stop-solution that includes risk assessment, source prediction and dashboards altogether so that the companies can invest their time worrying about right threats and get a far greater chance to beat the possible data breach.

Unbiased Risk Assessment:

In current scenario security solution providers give biased risk analysis so as to promote and sell their own products to the companies seeking protection against data breach. It may result into consideration of less effective security tools and technologies to analyze and protect the data from data breaching.
So, we are trying to provide a clean and straightforward risk analysis of data breach without promoting any product to help organizations protect their data in a better way.

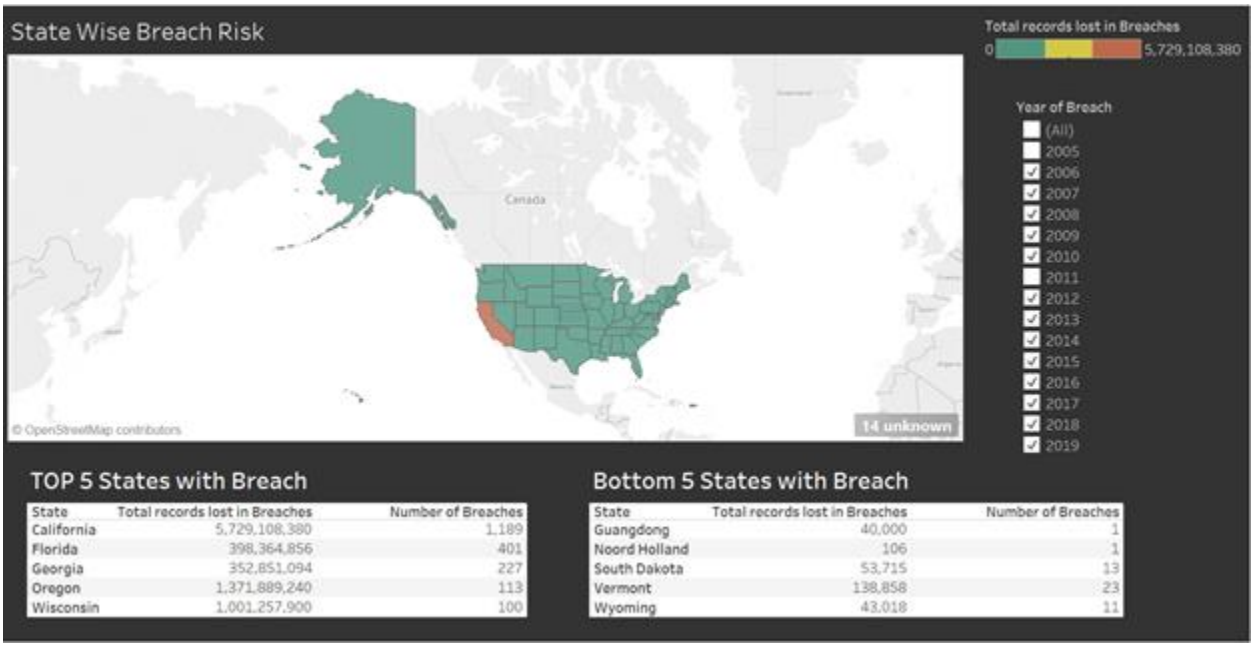
Comprehensible visualizations:
Risk analysis provided by companies lacks an interactive Dashboard for end user to analyze the changing dependency of different parameters affecting their data and susceptibility.
Permutation and combination of different factors and their outcome with lucid representation through an interactive dashboard is one of the prime features of our project implementation.

16. Directory description



17. Dashboards and insights

17.1 Tableau Dashboard 1 – State wise Breach Risk indication for all the US states

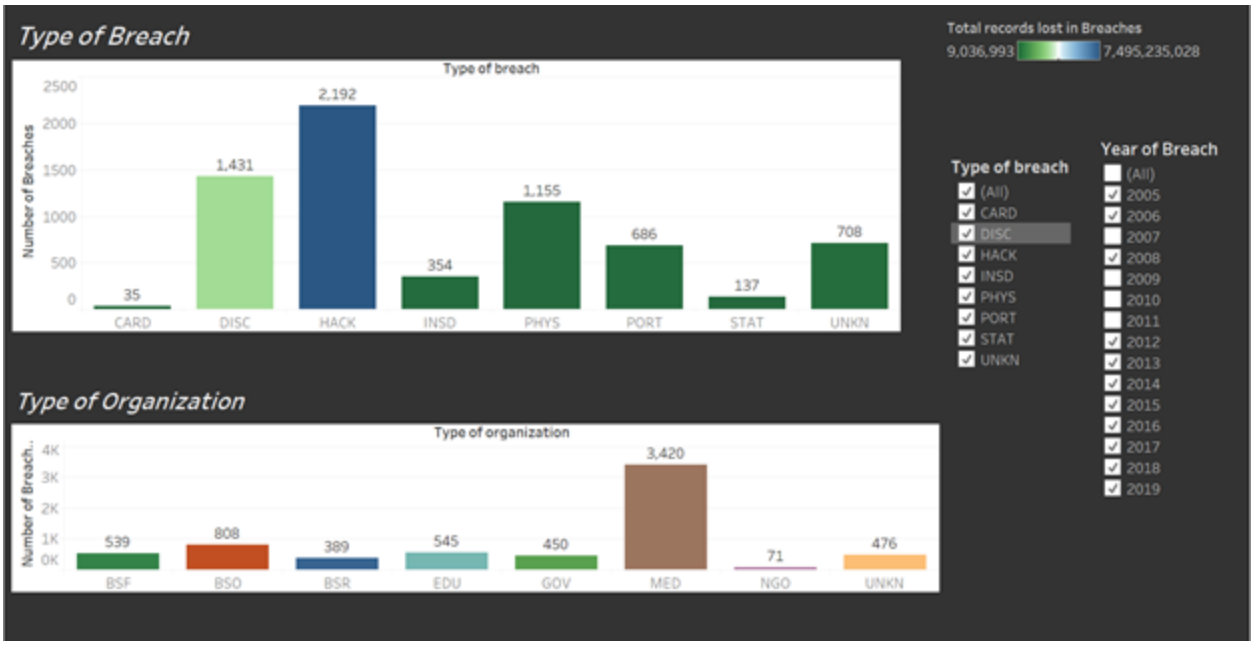


Dashboard comprises of:

- State wise Breach Risk Levels
- TOP 5 States with Breach
- Bottom 5 States with Breach
- Filter – Year

With the use of 'Actions' in Tableau, we integrated the sheet showing Top 5 States with Breach and Bottom 5 States with Breach. We used YEAR as a filter for the sheets and the year wise breach risk level was indicated across the states in the US.

17.2 Tableau Dashboard 2 – Type of Breach and the Organization

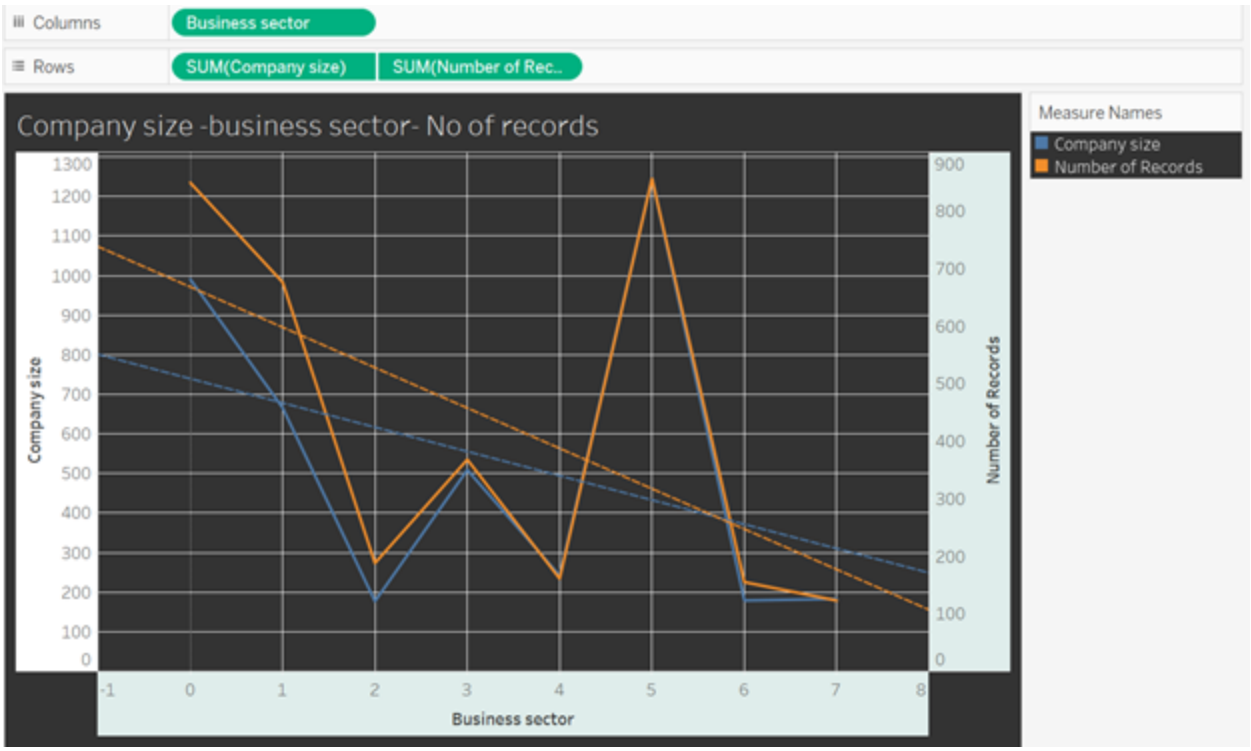


Dashboard comprises of –

- Number of Records for each Type of Breach
- Number of Records for each Type of Organization
- Filter – Year
- Filter – Type of Breach

The sheet of the number of records for the type of breach and type of organization were put on a dashboard with the help of filters and Actions in Tableau. This visualization helps in the low-level information as what is the number of times the breach occurred in an organization in a particular given year.

17.3 Tableau Dashboard 3 – Company Size – Business Sector and Number of Records



We have used the measure names for the company size and the number of records over the business sectors. With the help of the Trend lines we have tried to show the pattern for the business sector over the entire data.

18. Testing

18.1 Dashboards testing:

1. The URL provided on the Website should navigate to the provided dashboard.
2. Checked all the available data filters on the dashboard by applying them and check they are filtering properly
3. Checked and Validated if all the actions and dropdown features are working as expected
4. Checked if the export feature for exporting the Dashboard from Tableau desktop to tableau public is working as expected.
5. Checked if tableau public asks for credentials once navigated to the Dashboard exported from Tableau Desktop. Tableau public being an open visualization platform does not ask for login credentials and is open for usage.

18.2 Web application testing:

1. Users should have access to the services provided by the website (Risk assessment calculator, Interactive dashboard).
2. Checked whether all links on pages navigate to the desired web pages. Conformance to navigation flow is determined.
3. Checked for the error messages and unexpected system behavior in case of an unusual activity.
4. Checked whether the input from the user is captured by the system and output is displayed accordingly.
5. Checked for browser compatibility of web application with different web browsers including Google Chrome, Mozilla Firefox and Microsoft Edge.

18.3 Integrated system testing:

1. Checked if the Web application developed using Flask works correctly when launched.
2. Checked if the model execution is triggered with the submission of form details.
3. Versions for python and Flask and their compatibility is checked for the backend to run properly.
4. Checked if the pickled model integrated with the web application using Flask runs and returns an appropriate value on submitting input.
5. Checked if the dashboard is launched on tableau public as soon as the user navigates to the interactive dashboard page and clicks on the link for viewing the dashboard.
6. Checked if the system can navigate back to the website from the form submission page to home page and also from tableau public to the BreachEscape website.

19. Future Enhancements

1. Session control

We plan on implementing session control as future enhancement to our website. If a user is inactive for a long time on a particular webpage then the user will be notified of their inactivity and the session will expire.

2. Automated Recommendation system

We plan to include the functionality to propose the recommendations for a given user input regarding the activities in the workplace. This input could be a combination of factors. We plan to leverage Machine Learning models for the same. This functionality will provide users the ability to determine the solution to their predicted risk more confidently and maximize their chances to avoid breach events in future.

3. Developing the dashboard on some visualization rendering technology instead of visualization platform like tableau

Us being new to web programming, we are using Tableau as a platform for the visualization and creating dashboards. With further training and investing time in technologies like D3.js, we plan to implement all the visualizations in our web application itself so that the Dashboard is dynamically updated and there is no need of hosting tableau.

20. Learning Goals

The ultimate aim of this project is to develop an information system that solves a business problem and is commercially usable.
Gaining a real time experience by practically working on a business problem and infusing the concepts learned in classroom throughout MSIS program is the major learning goal.
We believed that our goals would be parallel to gaining knowledge and developing a system that would help us learn new things that would not have been possible in a classroom learning setup.
Our learning goals consists of:

1. Analyzing issues in data-related business environment and researching on such data.
2. Identifying problems and deriving insights and solutions.
3. Researching on the ways to solve the business problems.
4. Developing a working prototype of the website as a solution to the formulated business problem
5. Analyzing how this solution, in future, would thrive in the market when implemented in real-time

20.1 Personal learnings

Anurag Gate
During this project, I came across and learned few new technologies like web development using Flask for example. I got hands on experience on building a fully working interactive website using HTML5, CSS3, Python Flask. As we implemented the project with keeping in mind the principles of SDLC, I got a wonderful experience of Agile methodology throughout the project time. I also learned few interesting things while designing the Tableau dashboards for representing our data in a meaningful manner. Courses covered during my master's in information systems like DBMS, OOP, ML, Data science with python, ISAD, SPM, etc. helped me to apply my knowledge and design this planned project successfully.

Bhakti Mehta
I am currently pursuing my Masters in Information Systems at Santa Clara University. I have completed my Bachelors in Computer Engineering from University of Pune in 2016. As a part of my undergraduate course, I have studied Database Management Systems, Web Development and Software Engineering which helped me develop basic understanding regarding various topics we used in this project. I have an experience of a year at Cognizant Technology Systems in India. My Masters degree has helped me to gain deep knowledge and understanding about the field of Data Analytics - the track in which I aspire to pursue my future career.This project has helped me to develop my Data Science, Python and analytics skills. I used all phases of Data Science and Machine Learning. I worked in different parts of the Data Science Pipeline such as Data warehousing, Data extraction and pre – processing, training models and finally extracting insights from the ready and clean data.

Harshada Kulkarni
I am currently a student of Information Systems and an aspiring business analyst. The courses that I have completed in my undergraduate studies (Computer Science) and my masters have helped me gain in depth knowledge regarding the topics of Data Analysis, Information Systems Analysis, Process Design, Web Development, Software Design and Development. Working on the technical part of this project helped me in understanding implementation of Supervised and Unsupervised ML algorithms, Cross validation, Feature Engineering and Model Pickling. I learnt the concept of Flask, Efficiency metrics like ROC, Accuracy, confusion matrix and Tableau visualization thoroughly.

Prachi Tamhankar
Implementation of Agile methodology throughout the execution of the project and understand the entire project development life cycle.
Application of the course learnings during the MSIS program to implement the entire project.
Gained experience working on Machine learning using Python.
Learnt creation of web application using Flask.
Built Tableau dashboards to analyze the data for the number of records every year and present it to the users in an intuitive manner.

Sanjana Bothale
Currently pursuing Masters in Information Systems from Santa Clara University. I have two years of experience as a Quality Assurance Analyst with Accenture. I took courses like Information Systems Analysis and Design, Object oriented programming, Software project management, Database Management, Data Analysis which would help me in implementation of this project. With my work experience in Quality Assurance and curriculum here I planned to deliver a thoroughly tested quality product. Also from this group project I learnt different prediction algorithms based on their suitability for the data in hand. Also I got familiar with a framework-Flask. I could test a website developed for our project. The learning curve for me was steep.

21. References

https://breachlevelindex.com/

https://scotch.io/tutorials/authentication-and-authorization-with-flask-login

https://scikit-learn.org/stable/auto_examples/calibration/plot_compare_calibration.html#sphx-glr-auto-examples-calibration-plot-compare-calibration-py

https://data.gov.uk/dataset/64a0b145-6d34-4e53-a037-9dd5702074ce/information-security-breaches-survey

https://www.idtheftcenter.org/data-breaches/#top

https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/liu

https://towardsdatascience.com/designing-a-machine-learning-model-and-deploying-it-using-flask-on-heroku-9558ce6bde7b